

# Approximation algorithms for Euler genus and related problems

Chandra Chekuri\*

Anastasios Sidiropoulos†

April 10, 2013

## Abstract

The Euler genus of a graph is a fundamental and well-studied parameter in graph theory and topology. Computing it has been shown to be NP-hard by Thomassen [Tho89, Tho93], and it is known to be fixed-parameter tractable. However, the approximability of the Euler genus is wide open. While the existence of an  $O(1)$ -approximation is not ruled out, only an  $O(\sqrt{n})$ -approximation [CKK97] is known even in bounded degree graphs. In this paper we give a polynomial-time algorithm which on input a bounded-degree graph of Euler genus  $g$ , computes a drawing into a surface of Euler genus  $g^{O(1)} \cdot \log^{O(1)} n$ . Combined with the upper bound from [CKK97], our result also implies a  $O(n^{1/2-\alpha})$ -approximation, for some constant  $\alpha > 0$ .

Using our algorithm for approximating the Euler genus as a subroutine, we obtain, in a *unified* fashion, algorithms with approximation ratios of the form  $\text{OPT}^{O(1)} \cdot \log^{O(1)} n$  for several related problems on bounded degree graphs. These include the problems of orientable genus, crossing number, and planar edge and vertex deletion problems. Our algorithm and proof of correctness for the crossing number problem is shorter and simpler compared to the long and difficult proof in the recent (80 pages long) breakthrough by Chuzhoy [Chu11], while essentially obtaining a qualitatively similar result. For planar edge and vertex deletion problems our results are the first to obtain a bound of form  $\text{poly}(\text{OPT}, \log n)$ .

We also highlight some further applications of our results in the design of algorithms for graphs with small genus. Many such algorithms require that a drawing of the graph is given as part of the input. Our results imply that in several interesting cases, we can implement such algorithms even when the drawing is unknown.

---

\*Dept. of Computer Science, University of Illinois, Urbana, IL 61801. [chekuri@illinois.edu](mailto:chekuri@illinois.edu). Supported in part by NSF grant CCF-1016684.

†Dept. of Computer Science, University of Illinois, Urbana, IL 61801. [sidiropo@illinois.edu](mailto:sidiropo@illinois.edu). Supported in part by David and Lucille Packard Fellowship, NSF AF award CCF-0915984, and NSF grant CCF-0915519.

# 1 Introduction

A *drawing* of a graph  $G$  into a surface  $\mathcal{S}$  is a mapping that sends every vertex  $v \in V(G)$  into a point  $\varphi(v) \in \mathcal{S}$ , and every edge into a simple curve connecting its endpoints, so that the images of different edges are allowed to intersect only at their endpoints. In this paper we deal with closed surfaces (compact and without boundary). A surface  $\mathcal{S}$  can be orientable or non-orientable. The *Euler genus*  $\text{eg}(\mathcal{S})$  of a nonorientable surface  $\mathcal{S}$  is defined to be the nonorientable genus of  $\mathcal{S}$ . The Euler genus  $\text{eg}(\mathcal{S})$  of an orientable surface  $\mathcal{S}$  is equal to  $2\gamma$ , where  $\gamma$  is the orientable genus of  $\mathcal{S}$ . For a graph  $G$ , the Euler genus of  $G$ , denoted by  $\text{eg}(G)$ , is defined to be equal to the infimal Euler genus of a surface  $\mathcal{S}$ , such that  $G$  can be drawn into  $\mathcal{S}$ . The orientable genus of a graph  $G$ , denoted by  $\text{genus}(G)$ , is the infimal genus of an orientable surface  $\mathcal{S}$  into which  $G$  can be drawn.

Drawings of graphs into various surfaces are of central importance in graph theory (e.g. [GT01, MT01]), topology, and mathematics in general (e.g. [Whi01]), and have been the subject of intensive study. In particular, surface embedded graphs are an important ingredient in the seminal work of Robertson and Seymour on graph minors and the proof of Wagner’s conjecture. Surface embedded graphs are also important in computer science, and engineering, since they can be used to model a wide variety of natural objects.

We consider two simple-to-state and fundamental optimization problems in topological graph theory: given a graph  $G$ , compute  $\text{eg}(G)$  and  $\text{genus}(G)$ . Thomassen [Tho89] showed that computing these quantities is NP-hard. Deciding whether a graph has Euler genus 0, i.e. planarity testing, can be done in linear time by the seminal result of Hopcroft & Tarjan [HT74]. Deciding if  $\text{eg}(G) \leq g$  is fixed-parameter tractable. In fact, Mohar [Moh99] gave a linear time algorithm for this problem, and subsequently a relatively simple linear-time algorithm was given by Kawarabayashi, Mohar & Reed [KMR08]. The dependence of the running time in the above mentioned algorithms is at least exponential in  $g$ . We note that, for any fixed  $g$ , the set of all graphs with genus at most  $g$ , denoted by  $\mathcal{G}_g$  is minor-closed. From the work of Robertson and Seymour [RS90],  $\mathcal{G}_g$  is characterized as the class of graphs that exclude as a minor all graphs from a finite family of graphs  $\mathcal{H}_g$ . However,  $\mathcal{H}_g$  is not known explicitly even for small values of  $g$  and  $|\mathcal{H}_g|$  can be very large.

In this paper we consider the case when  $g$  is not a fixed constant and examine the *approximability* of  $\text{eg}(G)$  and  $\text{genus}(G)$ . Perhaps surprisingly, this problem is very poorly understood. We briefly describe the known results and illustrate the technical difficulties. In general,  $\text{eg}(G)$  can be as large as  $\Omega(n^2)$  where  $n$  is the number of nodes of  $G$  (e.g. for the complete graph  $K_n$ ), and Euler’s characteristic implies that any  $n$ -vertex graph of Euler genus  $g$  has at most  $O(n + g)$  edges. Since any graph can be drawn into a surface that has one handle for every edge, this immediately implies an  $O(n/g)$ -approximation, which is a  $\Theta(n)$ -approximation in the worst case. In other words, even though we currently cannot exclude the existence of an  $O(1)$ -approximation, the state of the art only gives a trivial  $O(n)$ -approximation. Using the fact that graphs of small genus have small balanced vertex-separators, Chen, Kanchi & Kanevsky [CKK97] obtained a simple  $O(\sqrt{n})$ -approximation for graphs of *bounded degree* which is currently the best known approximation ratio. In fact, if we do not assume bounded degree, nothing better than the trivial  $O(n/g)$ -approximation is known. Consider the case of apex graphs which are graphs that contain a single vertex whose removal makes them planar. Mohar [Moh01] showed that the genus problem for even these graphs is NP-hard. He also gave an elegant characterization of the genus for apex graphs, which can in turn be used to obtain a  $O(1)$ -approximation for such graphs. It is worth mentioning that essentially nothing is known for graphs with a constant number of apices! We also remark that by Euler’s formula, there is a trivial  $O(1)$ -approximation if the average degree is at least  $6 + \varepsilon$ , for some fixed  $\varepsilon > 0$ .

Finally, we mention a recent result by Makarychev, Nayyeri & Sidiropoulos [MNS12], who gave an algorithm that given a Hamiltonian graph  $G$  along with a Hamiltonian path  $P$ , draws the graph into a surface of Euler genus  $g^{O(1)} \log^{O(1)} n$  where  $g$  is the orientable genus of  $G$ . We note that their algorithm does not assume bounded degree which is its strength but assumes Hamiltonicity which is a limitation. Moreover, the techniques in [MNS12] rely heavily on using the given Hamiltonian path  $P$  while our techniques here are based on treewidth related ideas among several others.

Our algorithms for approximating genus also give us, in a *unified* fashion, algorithms for two related problems on drawing a graph on a planar surface, namely crossing number and planar edge/vertex deletion. The guarantees of our algorithms are of the form  $\text{OPT}^{O(1)} \log^{O(1)} n$ . These problems have also been well-studied and have the common feature that the known hardness results are weak (either NP-Hardness or APX-Hardness) while known approximation bounds are polynomial in  $n$  even in bounded-degree graphs. In this context we make some remarks on why the bounded-degree assumption is interesting despite being a limitation in some ways. First, we can assume that the graph has bounded average degree since otherwise the lower bound on the instance is very high and it becomes easy to approximate (see previous comment on genus). It is not uncommon in applications such as VLSI design and graph layout to assume some form of an upper bound on the degree; heuristically algorithms that work for bounded degree graphs can be extended to handle the case of graphs that can be made bounded degree by the removal of a small number of edges. Second, from a theoretical point of view, understanding the approximability even when all degrees are bounded (in fact at most 3) seems highly non-trivial and there has been very limited progress over two decades. It is only very recently that, in a long and difficult breakthrough work, Chuzhoy obtained a bound of the form  $\text{OPT}^{O(1)} \log^{O(1)} n$  for crossing number problem in bounded degree graphs. We now describe our results formally.

**Our results.** Our main result is an approximation algorithm for the Euler genus of bounded degree graphs. More specifically, given a graph  $G$  of Euler genus  $g$ , our algorithm computes a drawing of  $G$  into a surface of Euler genus  $\Delta^{O(1)} g^{O(1)} \log^{O(1)} n$  where  $\Delta$  is the maximum degree. The algorithm's running time is polynomial in both  $g$  and  $n$ . Combined with the simple  $O(n^{1/2})$ -approximation from [CKK97], our result gives a  $O(n^{1/2-\alpha})$ -approximation for some fixed constant  $\alpha > 0$ . The following theorem summarizes our main result.

**Theorem 1.1** (Main result). *There is a polynomial-time algorithm which given a graph  $G$  of maximum degree  $\Delta$ , and an integer  $g \geq 0$ , either outputs a drawing of  $G$  into a surface of Euler genus  $O(\Delta^3 g^7 \log^{13/2} n)$ , or correctly decides that the Euler genus of  $G$  is greater than  $g$ .*

**Remark 1.1.** *Kawarabayashi, Mohar and Reed [KMR08] obtain an exact algorithm to compute the Euler genus of a given graph in time  $2^{O(\text{OPT})} n$  time, which in particular implies a polynomial-time algorithm when  $\text{OPT} = O(\log n)$ ; this simplifies and improves a previous linear-time algorithm of Mohar [Moh99] which had a doubly-exponential dependence on  $\text{OPT}$ . Theorem 1.1, when combined with the algorithm in [KMR08], implies a polynomial-time algorithm that given a graph  $G$  outputs a drawing on a surface with Euler genus  $O(\Delta^3 \text{OPT}^{O(1)})$ .*

We build on our main result to obtain several other non-trivial results; we describe the outline of the unified methodology in Section 1.2. First, we obtain an algorithm to approximate  $\text{genus}(G)$ , the orientable genus of a given graph  $G$ , summarized in the theorem below. Note that  $\text{genus}(G)$  can be  $\Omega(\sqrt{|V(G)|})$  even when  $\text{eg}(G) = O(1)$  [FHRR95].

**Theorem 1.2** (Approximating the orientable genus). *There exists a polynomial-time algorithm which given a graph  $G$  of maximum degree  $\Delta$ , and an integer  $g > 0$ , either correctly decides that  $\text{genus}(G) > g$ , or outputs a drawing of  $G$  into a surface of orientable genus  $O(\Delta^4 g^9 \log^{13/2} n)$ .*

*Crossing number.* In the crossing number problem the input is a graph  $G$  which may not be planar and the goal is to draw it into the Euclidean plane with as few edge crossings as possible. When we deal with this problem, we will allow the edges in a graph drawing to intersect in their interiors. The point where the interiors of two edges intersect, is called a *crossing* of the drawing. We do not allow the interiors of edges to intersect vertices, and we also assume that there are no three edges, with their interiors intersecting at the same point. The *crossing number* of a graph  $G$ , denoted by  $\text{cr}(G)$ , is defined to be the smallest integer  $k$ , such that  $G$  admits a drawing into the plane, with at most  $k$  crossings.

The crossing number problem has also been a difficult problem to approximate, and the focus has been primarily on bounded degree graphs. It is an NP-hard problem but for each fixed  $k$  there is a linear time algorithm to decide if  $\text{cr}(G) \leq k$  [KR07]. In a recent breakthrough paper, Chuzhoy [Chu11] described an algorithm that given a graph  $G$  outputs a drawing into the plane with  $O(\text{cr}(G)^{10} \text{poly}(\Delta \log n))$  crossings; as a corollary she obtained the first algorithm that had an approximation ratio that is sub-linear in  $|V(G)|$ . The algorithm and proof in [Chu11] occupy almost 80 pages. It is a simple observation that if the crossing number of a graph  $G$  is  $k$  then  $\text{genus}(G) \leq k$  since one can add a handle for each edge that participates in a crossing. We can apply our approximation algorithm to find a drawing of  $G$  into an orientable surface, via Theorem 1.2, of genus  $O(\Delta^4 k^9 \log^{13/2} n)$ . Interestingly, having a drawing on a relatively low genus surface, allows us to obtain a rather simple algorithm for crossing number. Our result is summarized below.

**Theorem 1.3** (Approximating the crossing number). *There exists a polynomial-time algorithm which given a graph  $G$  of maximum degree  $\Delta$ , and an integer  $k \geq 0$ , either correctly decides that  $\text{cr}(G) > k$ , or outputs a drawing of  $G$  into the plane with at most  $O(\Delta^{11} k^{20} \log^{13} n)$  crossings.*

We note that the dependence on  $k$  in our theorem is worse than that in [Chu11]. However, we believe that our approach, in addition to giving a simpler proof, is interesting because it appears to differ from that in [Chu11] in going via a somewhat indirect route through a low genus drawing. We refer the interested reader to [CMS11, Chu11, CH11] for various pointers to the extensive work on crossing number and related problems.

*Planar Edge and Vertex Deletion.* We extend our approach via genus to obtain an approximation algorithm for the minimum planar edge/vertex deletion problems. In these problems we are given a graph  $G$  and the goal is to remove the smallest number of edges/vertices to make it planar. We denote by  $\text{edge-planarization}(G)$  the minimum size of such a planarizing set of edges and similarly by  $\text{vertex-planarization}(G)$  for vertices. The best known approximation for this problems is  $O(\sqrt{n} \log n)$  due to Tragoudas via the separator algorithms [LR99], and recently Chuzhoy [Chu11] gives an algorithm that outputs a solution of size  $O(\text{cr}(G)^5 \text{poly}(\Delta \cdot \log n))$ ; we observe that the  $\text{cr}(G)$  could be  $\Omega(\sqrt{n})$  even though there may be a single edge  $e$  such that  $G - e$  is planar. We obtain the first non-trivial approximation algorithm for this problem. Our result is summarized in the following Theorem.

**Theorem 1.4** (Approximating the minimum planar edge/vertex deletion). *There exists a polynomial-time algorithm which given a graph  $G$  of maximum degree  $\Delta$ , and an integer  $k > 0$ , either correctly decides that  $\text{edge-planarization}(G) > k$ , or outputs a set  $Y \subseteq E(G)$ , with  $|Y| = O(\Delta^6 k^{10} \log^{13/2} n)$ ,*

such that  $G \setminus Y$  is planar. Similarly, there is a polynomial-time algorithm that either correctly decides that  $\text{vertex-planarization}(G) > k$  or outputs a set  $X \subset V$  with  $|X| = O(\Delta^5 k^{10} \log^{13/2} n)$  such that  $G \setminus X$  is planar.

**Remark 1.2.** Our approach via genus gives algorithms with ratios  $O(\Delta^{O(1)} \text{OPT}^{O(1)})$  for crossing number and planar edge/vertex deletion. It is useful to note that, unlike for genus, crossing number and planar edge/vertex deletion do not yet have fixed-parameter-tractable algorithms that have a singly-exponential dependence on  $\text{OPT}$ .

**Further algorithmic applications.** Our approximation algorithm for Euler genus has further consequences in the design of algorithms for problems on graphs of small genus. Most algorithms that take advantage of the fact that a graph can be drawn on a surface of small genus require a drawing of the input graph be given as part of the input. If the genus  $g = O(\log n)$  then one can use existing exact algorithms that run in  $2^{O(g)} \text{poly}(n)$  time to obtain a drawing. Our result implies that we can obtain a drawing even when  $g = \Omega(\log n)$ , that while not being optimal, nevertheless yields interesting results. A concrete example of this application is the following. Recently, Erickson and Sidiropoulos [ES13] have obtained a  $O(\log g / \log \log g)$ -approximation for Asymmetric TSP on graph of Euler genus  $g$ , when a drawing of the graph is given as part of the input; this improves the bounds of Oveis-Gharan and Saberi [OS11] who gave an  $O(\sqrt{g} \log g)$ -approximation and also required the drawing as an input. Our result implies the following corollary: There exists a polynomial-time  $O(\log g / \log \log g)$ -approximation for ATSP on bounded-degree graphs of genus  $g$ , even when a drawing of the graph is not given as part of the input<sup>1</sup>.

The proof of Theorem 1.1 is somewhat technical and uses several ingredients. To aid the reader we first give an overview of the algorithmic ideas and highlight the ingredients that are needed. We assume that the reader is familiar with the notion of the treewidth of a graph. Section 1.2 highlights the high-level idea that allows us to leverage an algorithm for Euler genus for the other problems considered in the paper.

## 1.1 Overview of the algorithm for Euler genus

It is convenient to work with a promise version of the problem where we assume that  $\text{eg}(G)$  is at most a given number  $g$ . This allows us to assume certain properties that  $G$  needs to satisfy. Our algorithm may find that  $G$  does not satisfy such a property in which case it obtains a certificate that  $\text{eg}(G) > g$ .

**An idea from exact algorithms.** Our algorithm is inspired by fixed-parameter algorithms that run in polynomial time for any fixed genus [RS90, Moh99, KMR08]. It is instructive to briefly describe how such algorithms work. Let  $G$  denote the input graph, and suppose we want to find a drawing into a surface of Euler genus  $g$ , if one exists. If  $G$  happens to have bounded treewidth, say  $f(g)$  for some function  $f$ , then one can compute its Euler genus exactly via a dynamic program, in time roughly  $2^{O(f(g))} n^{O(1)}$ . If on the other hand  $G$  has treewidth larger than  $f(g)$ , by choosing  $f$  to be sufficiently large, a theorem of Robertson, and Seymour [RS86, RS03], asserts that  $G$  contains

---

<sup>1</sup>More precisely, there exists a polynomial-time algorithm which given a bounded-degree graph  $G$  (the instance of ATSP), and an integer  $g$ , either correctly decides that  $\text{eg}(G) > g$ , or it outputs a  $O(\log g / \log \log g)$ -approximate TSP tour in  $G$ .

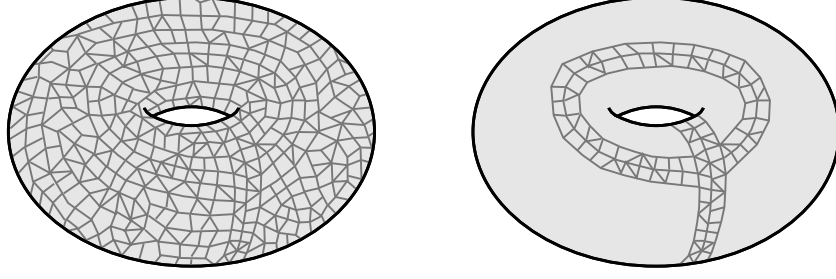


Figure 1: A high-treewidth graph drawn into the torus (left), and the low-treewidth skeleton obtained after removing irrelevant vertices (right).

a *flat*  $((2g+1) \times (2g+1))$ -grid minor  $H$ . Here, being flat means that the graph  $H$  admits a planar drawing, such that all edges leaving  $H$  are incident to the outer face. The central vertex  $v$  of such a grid can be shown to be *irrelevant*, i.e. such that  $G$  admits a drawing into a surface of Euler genus  $g$ , if and only if  $G - v$  does. Therefore, we can simply remove  $v$  from  $G$ , and recurse on the remaining graph. We continue removing irrelevant vertices in this fashion, until the treewidth becomes at most  $f(g)$ . We call the resulting low-treewidth graph a *skeleton* of  $G$  (see Figure 1). After drawing the skeleton, we can extend the drawing to all the removed irrelevant vertices.

**Challenges when  $g$  is not a fixed constant.** Our algorithm is based on modifying the above approach, so that it works in the approximate setting when  $g$  is part of the input. We now briefly describe the main challenges towards this goal. Let us begin with considering the case of a bounded-degree graph  $G$  of small treewidth, say at most  $g^{O(1)}$ , where  $g$  is the Euler genus of  $G$ . By repeatedly cutting along balanced separators, we can compute in polynomial time a set of at most  $\Delta^{O(1)} g^{O(1)} \log^{O(1)} n$  edges  $E^* \subset E(G)$ , such that  $G \setminus E^*$  is planar. By introducing one handle for every edge in  $E^*$ , we get a drawing of  $G$ , into a surface of Euler genus (in fact orientable genus)  $\Delta^{O(1)} g^{O(1)} \log^{O(1)} n$ .

Let us now consider the general case when treewidth of the graph  $G$  is larger than  $g^c$  for some sufficiently large constant  $c$ . Let us assume for now that we can again find an irrelevant vertex in  $G$ . It might seem at first that we are done, by proceeding as in the exact case and recursing on the reduced instance. However, this is the critical point where things break down in the approximate setting. Suppose that we remove a set  $U \subset V(G)$  of irrelevant vertices, such that the skeleton  $G \setminus U$  has treewidth  $g^{O(1)}$ . We know that the skeleton can be embedded with genus  $g$  iff  $G$  can. However, we only have an approximate algorithm for handling a low-treewidth graph. Using such an algorithm, we can compute a drawing  $\varphi$  of  $G \setminus U$  into a surface of Euler genus  $\Delta^{O(1)} g^{O(1)} \log^{O(1)} n$ . Unfortunately, now we are stuck! Since the drawing  $\varphi$  is not into a surface of Euler genus  $g$ , there might be no way of extending  $\varphi$  to  $U$ .

**Ensuring extendability.** We overcome the above issue by carefully computing irrelevant parts, that have some extra structure. This structure guarantees that the resulting approximate drawing of the skeleton can be extended to the whole  $G$ , by introducing only a small number of additional handles. To that end, we define a structure that we call a *patch*. A patch is simply a subgraph  $X \subset G$ , together with a cycle  $C$ , which we can think of as its “boundary”. We also think of  $X \setminus C$

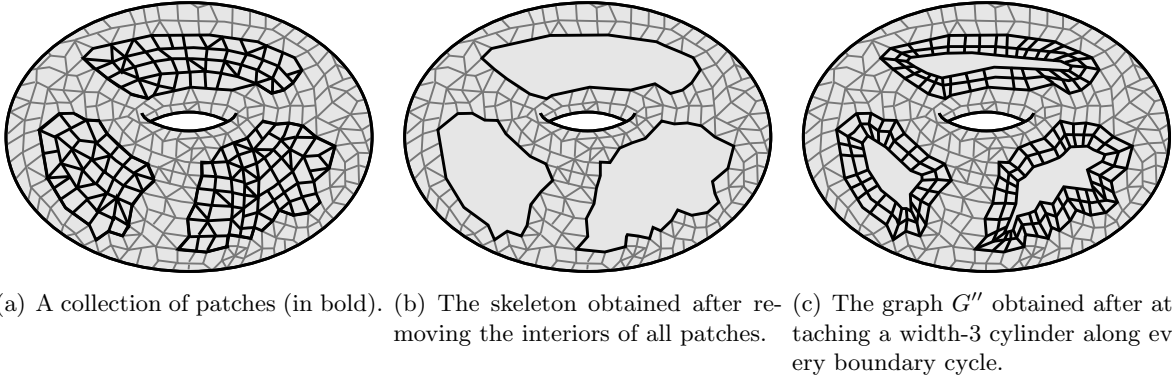


Figure 2: Constructing a skeleton by removing patches.

as the “interior” of the patch. Our goal is to compute patches  $X_1, \dots, X_k$ , satisfying the following two conditions:

- (C1) After removing the interiors of all patches, the resulting skeleton has treewidth at most  $g^{O(1)}$ .
- (C2) There exists a drawing  $\varphi_{\text{OPT}}$  of  $G$  into a surface  $\mathcal{S}$  of genus  $\text{eg}(G)$ , such that every patch  $X_i$  is drawn inside a disk  $\mathcal{D}_i$ , with its boundary being mapped to the boundary of  $\mathcal{D}_i$ . Moreover, the disks  $\mathcal{D}_i$  have pairwise disjoint interiors, and there is no part of  $G$  drawn inside each  $\mathcal{D}_i$ , other than  $X_i$  (see Figure 2(a)). We remark that we do not explicitly know  $\varphi_{\text{OPT}}$ , but we can nevertheless guarantee its existence.

Let us suppose for now that we can compute such a skeleton, with a corresponding collection of patches  $X_1, \dots, X_k$ . Let  $C_i$  be the boundary cycle of each  $X_i$ . Let  $G'$  be the skeleton  $G \setminus \bigcup_{i=1}^k (X_i \setminus C_i)$  (see Figure 2(b)). Let us now revisit the algorithm for low-treewidth graphs: We repeatedly remove balanced separators of size  $g^{O(1)} \log^{O(1)} n$ , until all connected components become planar. After removing a set  $E'$  of at most  $g^{O(1)} \log^{O(1)} n$  edges, we end up with a planar graph  $H' = G' \setminus E'$ . Fix a planar drawing  $\varphi'$  of  $H'$ . We would like to extend  $\varphi'$  to a low-genus drawing of the whole  $G$ . To that end, ideally, we would like every cycle  $C_i$  to bound a face in  $\varphi'$ . There are two things that can go wrong:

- (P1) A cycle might be broken into several different paths.
- (P2) A maximal segment  $P$  of a cycle  $C_i$  in  $H'$  might not be “one-sided”. That is, there might be no face of  $\varphi'$  containing  $P$  as a subpath.

Problem (P1) can be easily addressed: If a cycle gets broken into  $t$  pieces, then we can “fix” this by adding at most  $t$  extra handles. Since we remove only a small number of edges, and every edge can break at most two cycles, it follows that we only need to add a small number of new handles because of (P1).

Overcoming problem (P2) is somewhat more difficult: Intuitively, while computing the drawing of the skeleton  $G'$ , we modify  $G'$  by attaching a cylinder of width 3 on each cycle  $C_i$  (see Figure 2(c)). This ensures that in the resulting planar drawing of  $H'$ , each segment of every cycle is one-sided. In reality, things are more complicated, but this is the high-level idea. After computing

a planar drawing  $\varphi'$  as above, where every segment of a cycle is one-sided, we can extend  $\varphi'$  to a low-genus drawing of  $G$ .

**Computing the skeleton.** The missing ingredient is an algorithm to compute the skeleton satisfying the conditions described above. The challenging part is to satisfy condition (C2). One difficulty is that we can only compute patches iteratively. Hence, if we compute the patches naively, it is possible that a patch can “interfere” with previous patches. We avoid this by showing that every new patch, either is interior-disjoint from all previous ones, or it contains some of them in its interior. In the latter case, we can simply merge all internal patches with the current one. This is the technical part of the paper. Our proof uses several tools from the theory of graph minors, and topological graph theory, such as the grid-minor/treewidth duality, Whitney flips, and results on the so-called *planarly-nested sequences* [Moh92].

## 1.2 Orientable genus, Crossing number and Planar edge/vertex deletion

Our algorithms for  $\text{genus}(G)$ ,  $\text{cr}(G)$ ,  $\text{edge-planarization}(G)$  and  $\text{vertex-planarization}(G)$  rely on the algorithm for  $\text{eg}(G)$ . Interestingly having a drawing (even if it is into a non-orientable surface) helps via the following conceptually simple methodology. First we consider the problem of computing  $\text{genus}(G)$ . Suppose we have a drawing  $\varphi$  of  $G$  into a surface  $\mathcal{S}$  whose Euler genus is  $g^{O(1)} \log^{O(1)} n$  where  $g$  is  $\text{genus}(G)$ . Note that  $\text{eg}(G) \leq \text{genus}(G)$  and hence  $\text{eg}(G)$  provides a lower bound for  $\text{genus}(G)$ . We can efficiently check if  $\mathcal{S}$  is orientable or non-orientable. If  $\mathcal{S}$  is orientable then we are done. Suppose not. Then we compute  $\rho$ , the *representativity* (equivalently facewidth) of the drawing  $\varphi$  which captures how “densely”  $G$  is embedded in the surface — see Section 8 for a formal definition. If  $\rho$  is “small” relative to  $g$  we can cut a small number of edges along a non-contractible cycle and reduce the genus of the surface. We repeat this process until we obtain a drawing into a surface  $\mathcal{S}'$ , such that either  $\mathcal{S}'$  is orientable, or  $\mathcal{S}'$  is nonorientable, and the representativity is “large”. If  $\mathcal{S}'$  is orientable then we can add handles for all the edges cut along the way and obtain a drawing of the original graph into an orientable surface. The interesting case is when  $\mathcal{S}'$  is non-orientable and has high representativity. In this case we can show via results in [BMR96] that  $G$  has a large Möbius grid minor that certifies that  $\text{genus}(G) > g$ .

A similar approach works for  $\text{cr}(G)$  and  $\text{edge-planarization}(G)$ . It is an easy observation that for each of these problems we have  $\text{OPT} \geq \text{genus}(G)$  where  $\text{OPT}$  is the optimum value for the corresponding problem. Thus we can use our algorithm for  $\text{genus}(G)$  to first obtain an embedding into an orientable surface of genus comparable to  $\text{OPT}$ . We once again use the idea of representativity. Either we can iteratively keep cutting along short non-contractible cycles to reduce the genus by at least one in each step and obtain a planar graph, or we get stuck with an embedding on a non-planar surface with large representativity. In the latter case we find a certificate that  $\text{OPT}$  is large. In the former case we need to handle the small number of edges removed to obtain the planar graph. There is nothing to do for planar edge deletion since they are part of the output. For crossing number we can add these edges to the planar graph without incurring too many crossings via the results in [CMS11, CH11].

*Discussion:* One could argue that the main reason for the difficulty in approximating graph drawing problems is to get a suitable lower bound on the optimum value. Previous algorithms were based on divide and conquer based approach [BL84, LR99, CKK97]. However, this approach incurs an additive term that depends on the size of a graph and therefore one only obtains a polynomial-



factor approximation. On the other hand the problems are fixed parameter tractable so when  $\text{OPT}$  is quite small, one can obtain an exact algorithm. Chuzhoy's algorithm for crossing number, and our results, address the intermediate regime when  $\text{OPT}$  is not too small but is not so large that an additive term that depends on  $n$  can be ignored. Chuzhoy's algorithm and analysis are technically very involved but in essence her algorithm finds large rigid substructures in the given graph (via well-linked sets and grid minors) that have to be necessarily planar in any drawing with crossing number at most  $\text{cr}(G)$ . Our algorithms for crossing number and planar edge/vertex deletion, are indirect in that they are based on algorithms for genus. Consequently, the bounds we obtain are quantitatively somewhat weaker than those of Chuzhoy for crossing number. However, our algorithm offers a different perspective and approach which we believe is more transparent and easier to understand. We hope this will lead to a better understanding of the problem complexity and to improved algorithms.

### 1.3 Organization

The rest of the paper is organized as follows. In Section 2 we introduce some basic definitions. In Section 3 we give a procedure for simplifying the input graph. In Section 4 we present our algorithm for approximating the Euler genus, assuming an algorithm for computing the skeleton. In Section 5 we give the algorithm for computing the skeleton, assuming an algorithm for computing patches. The computation of patches uses as a subroutine an algorithm for computing flat grid minors, which is described in Section 6. The actual algorithm for computing patches is given in Section 7. Finally, our approximation algorithms for orientable genus, crossing number, and planar edge/vertex deletion are given in Sections 8, 9 and 10 respectively.

## 2 Preliminaries

For an orientable surface  $\mathcal{S}$ , let  $\text{genus}(\mathcal{S})$  denote its orientable genus. Similarly, for a graph  $G$ , let  $\text{genus}(G)$  denote its orientable genus. For a graph  $G$ , and for  $X, Y \subseteq V(G)$ , we use  $E(X, Y)$  to denote the set of edges with one end point in  $X$  and the other in  $Y$ . For  $X \subseteq V(G)$  we use  $N_G(X)$  to denote the neighbors of  $X$ , namely the set of vertices in  $V(G) \setminus X$  that have an edge to some vertex in  $X$ .

A graph  $H$  is a minor of a graph  $G$  if it is obtained from  $G$  by a sequence of edge deletions, edge contractions, and deletions of isolated vertices.

**Definition 2.1** (Minor mapping). *Let  $G$  be a graph, and let  $H$  be a minor of  $G$ . Then there exists a function  $\sigma : V(H) \rightarrow 2^{V(G)}$ , satisfying the following conditions:*

- (1) *For every  $v \in V(H)$ ,  $\sigma(v)$  induces a connected subgraph in  $G$ .*
- (2) *For any  $u \neq v \in V(H)$ , we have  $\sigma(u) \cap \sigma(v) = \emptyset$ .*
- (3) *For any  $\{u, v\} \in E(H)$ , there exist  $u' \in \sigma(u)$ , and  $v' \in \sigma(v)$ , such that  $\{u', v'\} \in E(G)$ .*

We refer to  $\sigma$  as a minor mapping (for  $H$ ). For a set  $U \subset V(H)$ , we will use the notation  $\sigma(U) = \bigcup_{v \in U} \sigma(v)$ .

**Definition 2.2** (Grids and cylinders). *Let  $r \geq 1$ ,  $k \geq 3$ . We define the  $(r \times k)$ -cylinder to be the Cartesian product of the  $r$ -path  $P$ , with the  $k$ -cycle  $C$ . We fix an endpoint  $v$  of  $P$ , and let  $u$  be the*

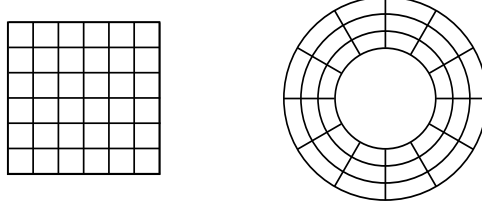


Figure 3: The  $(7 \times 7)$ -grid and the  $(4 \times 12)$ -cylinder.

other endpoint. We refer to the copy of the  $k$ -cycle  $\{v\} \times C$ , as the top, and to  $\{u\} \times C$ , as the bottom (of the cylinder).

Similarly, for  $s \geq 1$ ,  $t \geq 1$ , the  $(s \times t)$ -grid is the Cartesian product of the  $s$ -path  $P$ , with the  $t$ -path  $Q$ . We fix an endpoint  $v$  of  $P$ , and let  $u$  be the other endpoint. We refer to the copy of the  $t$ -path  $\{v\} \times Q$ , as the top, and to  $\{u\} \times Q$ , as the bottom (of the grid). See Figure 3.

We will make use of the following result of Feige et al. for computing balanced vertex-separators.

**Theorem 2.3** (Feige et al. [FHL08]). *There exists a polynomial time  $O(\sqrt{\log n})$ -pseudo approximation for balanced vertex separators. Moreover, given a graph  $G$  of treewidth  $t$ , we can compute in polynomial time a tree decomposition of  $G$  of width  $O(t\sqrt{\log t})$ .*

### 3 Graph normalization

Before we begin with the description of our algorithm, we give a procedure for simplifying the input graph. Throughout the proof of the main result, we will need to compute and maintain structures that satisfy certain properties in *any* optimal drawing. In order to achieve this, we need to enforce a certain type of “local rigidity” of drawings. To that end, it suffices to ensure that there are no planar components that can “flip” along a small vertex separator. The following is a formal definition of precisely this situation.

**Definition 3.1** (Freedom). *Let  $G$  be a graph, and let  $H \subseteq G$  be a vertex-induced subgraph of  $G$ . We say that  $H$  is free (in  $G$ ) if it satisfies the following conditions:*

- (1)  $|V(H) \cap V(G)| \leq 2$ . The vertices in  $V(H) \cap V(G)$  are called the portals of  $H$ .
- (2) If  $H$  has two portals  $t, t'$ , then  $H$  is not a path between  $t$  and  $t'$ .
- (3) There exists a planar drawing of  $H$  such that the portals lie in the boundary of the outer face.

If  $|V(H) \cap V(G)| = 1$ , we say that  $H$  is a petal, and if  $|V(H) \cap V(G)| = 2$ , we say that it is a clump.

**Definition 3.2** (Normalized graph). *We say that a graph  $G$  is normalized if there exists no free subgraph  $H \subseteq G$ .*

The following lemma allows us to restrict our attention to normalized graphs. A similar statement is proven in [Moh92].

**Lemma 3.3.** *Given a graph  $G$  of maximum degree  $\Delta$ , we can compute in polynomial time a graph  $G'$  of maximum degree at most  $\Delta$ , satisfying the following conditions:*

- (1) *The graph  $G'$  is normalized.*
- (2) *Given a drawing of  $G'$  into a surface  $\mathcal{S}$ , we can compute in polynomial time a drawing of  $G$  into  $\mathcal{S}$ .*

*Proof.* If  $G$  is normalized, then we can set  $G' = G$ . Otherwise, we start by computing an integer  $t \geq 0$ , and a sequence of graphs  $G = G_0, \dots, G_t$ . The graph  $G_t$  will be the desired normalized graph  $G'$ . Suppose we have computed  $G_i$ . If  $G_i$  is normalized, then we set  $t = i$ . Otherwise, we find  $X_i \subseteq V(G)$ , with  $|X_i| \leq 2$ , and such that some connected component  $C_i$  of  $G_i \setminus X_i$  is planar. If  $|X_i| = 1$ , then we set  $G_{i+1} = G_i \setminus (V(C_i) \setminus X_i)$ . Otherwise, if  $|X_i| = 2$ , we remove all vertices in  $V(G_i) \setminus X_i$ , and we add an edge  $e_i$  between the two vertices in  $X_i$ . Since  $|V(G_{i+1})| < |V(G_i)|$ , the above process terminates in polynomial time, with a normalized graph  $G_t = G'$ .

It suffices to show that given a drawing  $\varphi_{i+1}$  of  $G_{i+1}$  into a surface  $\mathcal{S}$ , we can compute in polynomial time a drawing  $\varphi_i$  of  $G_i$  into the same surface. Let  $H_i = G_i[X_i]$ , and let  $T_i = X_i \cap V(G_i)$  be the set of portals of  $H_i$ . Since  $H_i$  is free, it admits a planar drawing  $\psi_i$ , in which all its portals lie in the boundary of the outer face. Therefore, there exists a drawing  $\psi'_i$  of  $H_i$  into a disk  $\mathcal{D}_i$ , such that  $\psi'_i(H_i) \cap \partial\mathcal{D}_i = \psi'_i(T_i)$ . If  $H_i$  is a petal, then  $T_i = \{t\}$ . There exists a disk  $\mathcal{R}_i \subset \mathcal{S}$ , intersecting  $\varphi_{i+1}(G_{i+1})$  only on  $\varphi_{i+1}(t)$ . Embedding the disk  $\mathcal{D}_i$  onto  $\mathcal{R}_i$  results into the desired drawing for  $G_i$ . Similarly, if  $H_i$  is a clump, then there exists an edge  $e_i\{t_1, t_2\} \in E(G_{i+1})$ , where  $t_1, t_2$  are the two portals of  $H_i$ . There exists a disk  $\mathcal{R}'_i \subset \mathcal{S}$  such that  $\varphi_{i+1}(G_{i+1}) \cap \mathcal{R}'_i = \varphi_{i+1}(e_i)$ . Embedding the disk  $\mathcal{D}_i$  onto  $\mathcal{R}'_i$  results into the desired drawing for  $G_i$ . This concludes the proof.  $\square$

## 4 The algorithm

We begin by formally defining the notion of a patch, which we alluded to in Section 1.1.

**Definition 4.1** (Patch). *Let  $G$  be a graph. Let  $X \subseteq G$  be a subgraph, and let  $C \subsetneq X$  be a cycle. Then, we say that the ordered pair  $(X, C)$  is a patch (of  $G$ ).*

Note that the above definition of a patch is completely combinatorial, i.e. it is completely independent from drawings of the graph  $G$ . We will often refer to a patch, w.r.t. a specific drawing. This is captured in the following definition.

**Definition 4.2** ( $\varphi$ -Patch). *Let  $G$  be a graph, and let  $(X, C)$  be a patch of  $G$ . Let  $\varphi$  be a drawing of  $G$  into a surface  $\mathcal{S}$ . We say that  $(X, C)$  is a  $\varphi$ -patch (of  $G$ ), if there exists a disk  $\mathcal{D} \subset \mathcal{S}$ , satisfying the following conditions:*

- (1)  $\partial\mathcal{D} = \varphi(C)$ .
- (2)  $\varphi(G) \cap \mathcal{D} = \varphi(X)$ .

The following definition captures the notion of a pair of “interfering” patches.

**Definition 4.3** (Overlapping patches). *Let  $G$  be a graph, and let  $(X_1, C_1), (X_2, C_2)$  be patches of  $G$ . We say that  $(X_1, C_1)$ , and  $(X_2, C_2)$  are overlapping if either  $(X_1 \setminus C_1) \cap X_2 \neq \emptyset$ , or  $(X_2 \setminus C_2) \cap X_1 \neq \emptyset$ . In particular, if  $(X_1, C_1)$ , and  $(X_2, C_2)$  are non-overlapping, then this definition implies  $X_1 \cap X_2 = C_1 \cap C_2$ .*

Our general goal will be to compute patches that do not interfere precisely in the above sense. We now have all the notation in place, to state the main result for computing a skeleton of the input graph.

**Lemma 4.4** (Computing a skeleton). *There exists a polynomial-time algorithm which given a graph  $G$  of treewidth  $t \geq 1$ , and maximum degree  $\Delta$ , and an integer  $g > 0$ , either correctly decides that  $\text{eg}(G) > g$ , or outputs a collection of pairwise non-overlapping patches  $(X_1, C_1), \dots, (X_r, C_r)$  of  $G$ , so that the following conditions are satisfied:*

- (1) *If  $\text{eg}(G) \leq g$ , then there exists a drawing  $\varphi$  of  $G$  into a surface of Euler genus  $g$ , such that for any  $i \in \{1, \dots, r\}$ ,  $(X_i, C_i)$  is a  $\varphi$ -patch. We emphasize the fact that  $\varphi$  is not explicitly computed by the algorithm.*
- (2) *The graph  $G \setminus (\bigcup_{i=1}^r (X_i \setminus C_i))$  has treewidth at most  $O(\Delta^2 g^6 \log^5 n)$ .*

Lemma 4.4 is the main technical part of the paper. In the interest of clarity, we postpone its proof to later sections, and we instead show now how it can be used to obtain our approximation algorithm for Euler genus.

Before we describe the actual algorithm, we need to define a local “framing” operation, which we use to modify the skeleton. Intuitively, this is needed to ensure that when computing a drawing for the skeleton, the boundaries of the patches are drawn in a “nearly one-sided” fashion. This “near one-sidedness” in turn will allow us to extend the drawing of the skeleton, to a drawing of the whole graph. Note that framing is a combinatorial operation and does not rely on a drawing.

**Definition 4.5** (Graph framing). *Let  $G$  be a graph, and let  $\mathcal{C} = C_1, \dots, C_k \subseteq G$  be a collection of cycles. Let  $G'$  be the graph obtained from  $G$  by taking for every  $i \in \{1, \dots, k\}$ , a copy  $K_i$  of the  $(3 \times |V(C_i)|)$ -cylinder, and identifying the top of  $K_i$  with  $C_i$ . We refer to  $G'$  as the  $\mathcal{C}$ -framing of  $G$  (see Figure 4(a)).*

*More generally, we define the framing operation for subgraphs. Let  $H \subseteq G$  be a subgraph of  $G$ . We define a graph  $H'$  as follows. Consider some  $C \in \mathcal{C}$ . If  $C \subseteq H$ , then we take a copy of the  $(3 \times |V(C)|)$ -cylinder, and we identify its top with  $C$ . If  $C \not\subseteq H$ , then let  $P_1, \dots, P_a$  be the set of maximal subpaths of  $C$  that are contained in  $H$ . For every such  $P_j$ , we take a copy of the  $(3 \times |V(P_j)|)$ -grid, and we identify its top with  $P_j$ . We repeat this process for all  $C_i \in \mathcal{C}$ , and we define  $H'$  to be the resulting graph. We refer to  $H'$  as the  $\mathcal{C}$ -framing of  $H$  (see Figure 4(b)). The reader can check that the definition of the  $\mathcal{C}$ -framing of  $H$  agrees with the one given above, when  $H = G$ .*

We now state a basic property of framing whose proof follows directly from the definition of a  $\mathcal{C}$ -framing.

**Lemma 4.6.** *Let  $G$  be a graph, and let  $\mathcal{C}$  be a collection of cycles in  $G$ . Let  $\mathcal{H}$  be a collection of pairwise vertex-disjoint subgraphs of  $G$ . Let  $G'$  be the  $\mathcal{C}$ -framing of  $G$ , and for any  $H \in \mathcal{H}$ , let  $H'$  be the  $\mathcal{C}$ -framing of  $H$ . Then, the graph  $\bigcup_{H \in \mathcal{H}} H'$  is (isomorphic to) a subgraph of  $G'$ .*

We first argue that framing does not increase the genus of the skeleton.

**Lemma 4.7** (The genus of a framed skeleton). *Let  $G$  be a graph, and let  $\varphi$  be a drawing of  $G$  into some surface  $\mathcal{S}$ . Let  $\mathcal{P}$  be a collection of pairwise non-overlapping  $\varphi$ -patches of  $G$ . Let  $G' = G \setminus \left( \bigcup_{(X,C) \in \mathcal{P}} X \setminus C \right)$ . Let  $G''$  be the  $\{C\}_{(X,C) \in \mathcal{P}}$ -framing of  $G'$ . Then,  $\text{eg}(G'') \leq \text{eg}(G)$ .*

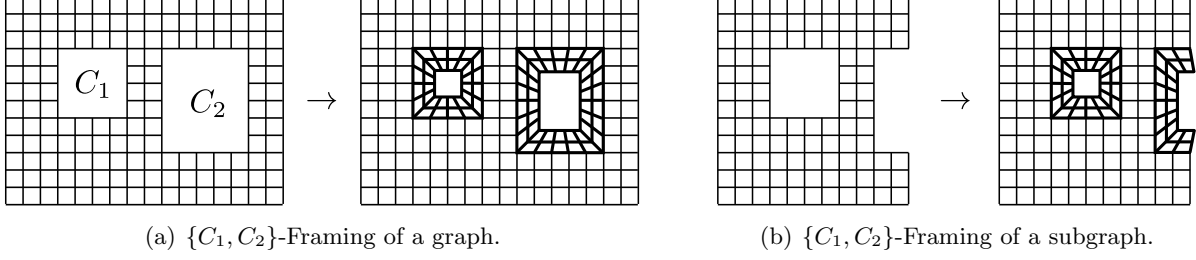


Figure 4: Examples of graph framing.

*Proof.* For every  $(X, C) \in \mathcal{P}$ , there exists a disk  $\mathcal{D}_{(X, C)} \subset \mathcal{S}$ , with  $\partial\mathcal{D}_{(X, C)} = \varphi(C)$ , and such that  $\varphi(G) \cap \mathcal{D}_{(X, C)} = \varphi(X)$ . Moreover, since the patches are pairwise non-overlapping, it follows that the resulting disks have pairwise disjoint interiors. For every patch  $(X, C) \in \mathcal{P}$ , there exists a cylinder  $K_{(X, C)} \subseteq G''$ , such that the top of  $K_{(X, C)}$  has been identified with  $C$ . We remove  $X \setminus C$  from the drawing  $\varphi$ , and we draw the rest of the cylinder  $K_{(X, C)}$  inside  $\mathcal{D}_{(X, C)}$ . Repeating for all patches in  $\mathcal{P}$ , we obtain a drawing of  $G''$  into  $\mathcal{S}$ . This shows that  $\text{eg}(G'') \leq \text{eg}(G)$ , which concludes the proof.  $\square$

**Lemma 4.8** (Planarizing the skeleton). *There exists a polynomial-time algorithm which given a graph  $G$  of maximum degree  $\Delta$ , an integer  $g > 0$ , and a collection of pairwise non-overlapping patches  $(X_1, C_1), \dots, (X_r, C_r)$  of  $G$  satisfying the assertion of Lemma 4.4, it either correctly decides that  $\text{eg}(G) > g$ , or it outputs a set  $S \subseteq V(G)$ , satisfying the following conditions:*

$$(1) |S| = O(\Delta^2 g^7 \log^{13/2} n).$$

(2) Let  $\mathcal{C} = \{C_1, \dots, C_r\}$ , and let  $G' = G \setminus (\bigcup_{i=1}^r (X_i \setminus C_i))$ . For every connected component  $H$  of  $G' \setminus S$ , we have that the  $\mathcal{C}$ -framing of  $H$  is planar.

*Proof.* By Lemma 4.4,  $G'$  has treewidth  $t' = O(\Delta^2 g^6 \log^5 n)$ . For a subgraph  $H \subseteq G'$ , let  $H^{\text{framed}}$  be the  $\mathcal{C}$ -framing of  $H$ . We proceed to compute an auxiliary tree  $\mathcal{T}$ , where every vertex of  $\mathcal{T}$  is a subgraph of  $G'$ , and we consider  $\mathcal{T}$  as being rooted at  $G'$ . We construct  $\mathcal{T}$  inductively as follows: For every vertex  $H \subseteq G'$  of  $\mathcal{T}$ , if  $H^{\text{framed}}$  is planar, then we set  $H$  to be a leaf of  $\mathcal{T}$ . Otherwise, using the algorithm from Theorem 2.3 we find in polynomial time a  $\Theta(1)$ -balanced vertex separator  $S_H \subseteq V(H)$ , with

$$|S_H| = O(t' \sqrt{\log n}) = O(\Delta^2 g^6 \log^{11/2} n).$$

For every connected component  $H'$  of  $H \setminus S_H$ , we add  $H'$  to  $V(\mathcal{T})$ , and we make  $H'$  a child of  $H$ . This concludes the definition of  $\mathcal{T}$ . Clearly,  $\mathcal{T}$  can be computed in polynomial time.

The height of  $\mathcal{T}$  is  $h = O(\log n)$  since we use  $\Theta(1)$ -balanced vertex separators to split each graph at an internal node of  $\mathcal{T}$ . For any  $i \in \{0, \dots, h\}$ , let  $\mathcal{L}_i$  be the set of vertices of  $\mathcal{T}$  that are at level  $i$  (i.e.,  $\mathcal{L}_0 = \{G'\}$ ). Note that all elements in any  $\mathcal{L}_i$  are pairwise vertex-disjoint subgraphs of  $G$ . It follows by Lemma 4.6 that for any  $i \in \{0, \dots, h\}$ ,

$$\bigcup_{H \in \mathcal{L}_i} H^{\text{framed}} \subseteq (G')^{\text{framed}}.$$

Note that all non-leaf subgraphs of  $G'$  in  $\mathcal{L}_i$  are non-planar. Therefore, by Lemma 4.7 there can be at most  $O(g)$  non-leaf subgraphs in  $\mathcal{L}_i$ . If this is not the case, then we can correctly decide that  $\text{eg}(G) > g$ . Otherwise, let  $S$  be the set of all vertex separators computed throughout the construction of  $\mathcal{T}$ , i.e.  $S = \bigcup_H S_H$ , where  $H$  ranges over all non-leaf vertices of  $\mathcal{T}$ . We have

$$\begin{aligned} |S| &= \sum_H |S_H| \\ &\leq \sum_{i=0}^h |\mathcal{L}_i| \cdot O(\Delta^2 g^6 \log^{11/2} n) \\ &= O(\Delta^2 g^7 \log^{13/2} n), \end{aligned}$$

as required.  $\square$

We can now prove the main result of this paper.

*Proof of Theorem 1.1.* By Lemma 4.4, in polynomial time, we can either correctly decide that  $\text{eg}(G) > g$ , or we can compute a (possibly empty) collection  $\mathcal{P}$  of pairwise non-overlapping patches of  $G$ , satisfying the following conditions:

- If  $\text{eg}(G) \leq g$ , then there exists a drawing  $\varphi$  of  $G$  into a surface  $\mathcal{S}$  of Euler genus  $g$ , such that any  $(X, C) \in \mathcal{P}$  is a  $\varphi$ -patch.
- The graph  $G' = G \setminus \left( \bigcup_{(X,C) \in \mathcal{P}} X \setminus C \right)$  has treewidth  $t' = O(\Delta^2 g^6 \log^5 n)$ .

Let  $\mathcal{C} = \{C : (X, C) \in \mathcal{P}\}$ . For any subgraph  $H \subseteq G'$ , let  $H^{\text{framed}}$  denote the  $\mathcal{C}$ -framing of  $H$ .

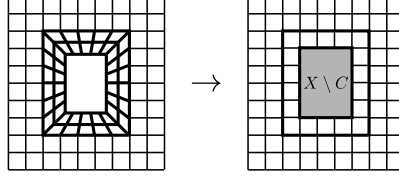
Let  $S \subseteq V(G)$  be the set computed by Lemma 4.8. Let  $G'' = G' \setminus S$ , and let  $\mathcal{H}$  be the set of connected components of  $G''$ . Since for every  $H \in \mathcal{H}$  the graph  $H^{\text{framed}}$  is planar, it follows that  $(G'')^{\text{framed}} = \bigcup_{H \in \mathcal{H}} H^{\text{framed}}$  is also planar. Pick a planar drawing  $\psi$  of  $(G'')^{\text{framed}}$  (which can be computed, e.g. by the algorithm of Hopcroft, and Tarjan [HT74]).

We now proceed to obtain a drawing of  $G$ , by modifying the drawing  $\psi$  of  $(G'')^{\text{framed}}$ . We iterate over all patches  $(X, C) \in \mathcal{P}$ . Consider some  $(X, C) \in \mathcal{P}$ . Since every  $(X, C) \in \mathcal{P}$  is a  $\varphi$ -patch, it follows that the graph  $X$  admits a planar drawing  $\varphi_{(X,C)}$  into a disk  $\mathcal{D}_{(X,C)}$ , with  $\varphi_{(X,C)}(C) = \partial\mathcal{D}_{(X,C)}$ . Let

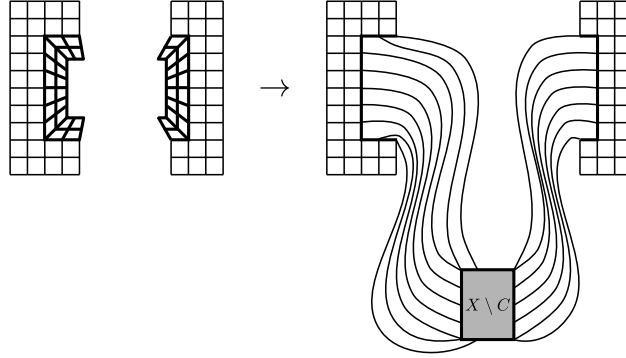
$$n_C = |E(C) \setminus E(G'')|.$$

We consider two cases:

- If  $n_C = 0$ , it follows that  $C \subseteq G''$ . Therefore,  $(G'')^{\text{framed}}$  contains a  $(3 \times |V(C)|)$ -cylinder  $K$ . The top of  $K$  is identified with  $C$  in  $(G'')^{\text{framed}}$ . Since  $K$  is 3-vertex-connected, it admits a unique planar drawing. Let  $C'$  be the bottom of  $K$ . It follows  $K$  bounds a face  $F$  in  $\psi$ . We can therefore extend the current drawing to  $X \setminus C$ , by placing the open disk  $\mathcal{D}_{(X,C)} \setminus \partial\mathcal{D}_{(X,C)}$  inside the face  $F$ , deleting all vertices of  $K$  that do not belong to its top, and connecting the vertices in  $X$ , with their neighbors in the copy of  $C$  in  $G''$ . Notice that in this case, we do not increase the genus of the current surface.



- (ii) If  $n_C > 0$ , we proceed as follows. First, we place the disk  $\mathcal{D}_{(X,C)}$  in the unbounded face of the current drawing. Let  $P_1, \dots, P_{n_C}$  be the set of maximal segments of  $C$  contained in  $G''$ . Consider a maximal segment  $P_j$ . There exists a  $(3 \times |V(P_j)|)$ -grid  $L$  in  $(G'')^{\text{framed}}$ , such that the top of  $L$  is identified with  $P_j$ . We argue that the bottom of  $L$  is a segment of a face  $F$  in  $\psi$ : If  $|V(P_j)| \leq 2$ , this is immediate, and if  $|V(P_j)| \geq 3$ , it follows by Whitney's theorem, since  $L$  is 3-connected, and therefore has a unique planar drawing. If the orientation of  $P_j$  along a clockwise traversal of  $\partial\mathcal{D}_{(X,C)}$  agrees with the orientation of  $P_j$  along a clockwise traversal of  $F$ , then we attach a handle between  $P_j$  in  $X$ , and the bottom of  $L$ . Otherwise, we attach a Möbius band. Next, we delete all the vertices in  $L$  that do not belong to its bottom, and we also delete the copy of  $P_j$  in  $X$  (i.e. the copy that lies on the boundary of  $\mathcal{D}_{(X,C)}$ ). Finally, we draw the edges between  $X \setminus C$ , and  $P_j$ , by routing them along the new handle, or Möbius band. We continue in the same fashion, with all the remaining maximal segments. For every maximal segment, we increase the Euler genus of the underlying surface by at most  $O(1)$ . Therefore, the total increase in the Euler genus is at most  $O(n_C)$ .



After considering all patches in  $\mathcal{P}$ , we arrive at a drawing into some surface. We remove any remaining vertices from  $(G'')^{\text{framed}} \setminus G''$ . We arrive at a drawing  $\psi'$  of the graph

$$\Gamma = G \left[ V(G'') \cup \left( \bigcup_{(X,C) \in \mathcal{P}} X \setminus C \right) \right] = G \setminus S.$$

Since the cycles  $\{\varphi(C)\}_{C \in \mathcal{C}}$  bound disks with disjoint interiors in the drawing  $\varphi$ , it follows that every edge of  $G'$  is contained in at most two cycles in  $\mathcal{C}$ . Therefore,

$$\begin{aligned} \sum_{C \in \mathcal{C}} n_C &\leq 2 \cdot |E(G') \setminus E(G'')| \leq \Delta \cdot |V(G') \setminus V(G'')| \\ &= \Delta \cdot |S| = O(\Delta^3 g^7 \log^{13/2} n). \end{aligned}$$

It follows that the resulting drawing  $\psi'$  of  $\Gamma$  is into a surface of Euler genus at most  $O(\sum_{C \in \mathcal{C}} n_C) = O(\Delta^3 g^7 \log^{13/2} n)$ .

It remains to extend the drawing to  $S$ . This can be done by adding at most  $|S| \cdot \Delta$  additional handles (one for every edge incident to a vertex in  $S$ ). The resulting drawing of  $G$  has Euler genus  $O(\Delta^3 g^7 \log^{13/2} n) + O(|S| \cdot \Delta) = O(\Delta^3 g^7 \log^{13/2} n)$ , as required. This concludes the proof.  $\square$

## 5 Computing a skeleton

The rest of the paper is devoted to the algorithm for computing a skeleton, i.e. the proof of Lemma 4.4. At the high level, the algorithm proceeds iteratively as follows: (i) We compute a patch. (ii) We remove its interior. (iii) We repeat until the treewidth of the remaining graph becomes small enough. One issue with implementing the above approach is that we do not have access to an explicit optimal drawing of the input graph. Therefore, we cannot argue that a computed patch is a  $\varphi$ -patch for a *specific* optimal drawing. To that end, we will need a stronger notion of a patch. More precisely, we need to compute patches that are  $\varphi$ -patches, in *any* optimal drawing  $\varphi$ . Moreover, because the above procedure computes patches in an ever decreasing graph, we need to make sure that the genus never decreases. This property will ensure that at the end of the procedure, all computed patches are  $\varphi$ -patches for *the same* optimal drawing  $\varphi$ . The following definition states precisely the properties that we need.

**Definition 5.1** (Universal patch). *Let  $G$  be a graph of Euler genus  $g$ . Let  $X \subseteq G$  be a subgraph of  $G$ , and let  $C \subsetneq X$  be a cycle in  $X$ . We say that  $(X, C)$  is a universal patch (of  $G$ ) if it satisfies the following conditions:*

- (1) *For any drawing  $\psi$  of  $G$  into a surface of Euler genus  $g$ , we have that  $(X, C)$  is a  $\psi$ -patch.*
- (2) *Let  $G' = G \setminus (X \setminus C)$ . Then,  $\text{eg}(G') = \text{eg}(G) = g$ .*

The following lemma shows that we can compute a universal patch in polynomial time, in a normalized graph of sufficiently large treewidth. A crucial property of the algorithm is that after removing the interior of the computed patch, the resulting graph remains normalized. This fact allows us to inductively maintain a normalized graph, while computing the skeleton. Intuitively, dealing with a normalized graph is essential for avoiding “locally-pathological” planar drawings. Roughly speaking, a non-normalized graph can have an optimal drawing that locally looks rather complicated. This makes it very difficult to control how different overlapping patches interact with each other.

**Lemma 5.2** (Computing a universal patch). *There exists a universal constant  $\alpha > 0$ , such that the following holds. Let  $G$  be a normalized graph of Euler genus  $g \geq 1$ , treewidth  $t \geq 1$ , and maximum degree  $\Delta$ . Suppose that  $t \geq \alpha \Delta^2 g^6 \log^{9/2} n$ . Then, we can compute in polynomial time a universal patch  $(X, C)$  in  $G$ , such that  $G \setminus (X \setminus C)$  is normalized.*

The proof of Lemma 5.2 is rather long, and requires several other results, including our algorithm for computing a flat grid minor, and properties of planarly nested sequences due to Mohar [Moh92]. For that reason, we defer it to subsequent sections, and show first how to use it to construct a skeleton (i.e. to prove Lemma 4.4).

Before we proceed with the proof of Lemma 4.4, we first derive a property that will be used in showing the correctness of the algorithm. While computing a sequence of patches, it is possible



that a patch overlaps a previously computed one. In this case, we can show that we can essentially “merge” the two patches. The following Lemma gives the necessary properties for doing exactly that.

**Lemma 5.3** (Merging overlapping patches). *Let  $G$  be a graph, let  $\varphi_1$  be a drawing of  $G$  into a surface  $\mathcal{S}$ , and let  $(X_1, C_1)$  be a  $\varphi_1$ -patch of  $G$ . Let  $G' = G \setminus (X_1 \setminus C_1)$ . Let  $\varphi_2$  be the drawing of  $G'$  into  $\mathcal{S}$  obtained by restricting  $\varphi_1$  to  $G'$ . Let  $(X_2, C_2)$  be a  $\varphi_2$ -patch of  $G'$ . Suppose further that  $(V(X_2) \setminus V(C_2)) \cap V(C_1) \neq \emptyset$ . Then,  $(X_1 \cup X_2, C_2)$  is a  $\varphi_1$ -patch of  $G$ .*

*Proof.* Since  $(X_2, C_2)$  is a  $\varphi_2$ -patch, it follows that there exists  $\mathcal{D}_2 \subset \mathcal{S}$ , such that  $\partial\mathcal{D}_2 = \varphi_2(C_2)$ . It remains to show that  $\varphi_1(G) \cap \mathcal{D}_2 = \varphi_1(X_1 \cup X_2)$ . Since  $(X_1, C_1)$  is a  $\varphi_1$ -patch, it follows that there exists a disk  $\mathcal{D}_1 \subset \mathcal{S}$ , with  $\partial\mathcal{D}_1 = \varphi_1(C_1)$ . We claim that  $\mathcal{D}_1 \subseteq \mathcal{D}_2$ . Suppose to the contrary that there exists a point  $p \in \mathcal{D}_1 \setminus \mathcal{D}_2$ . Pick an arbitrary vertex  $v \in (V(X_2) \setminus V(C_2)) \cap V(C_1)$ , and let  $q = \varphi_1(v) = \varphi_2(v)$ . We have  $q \in \mathcal{D}_2 \cap \mathcal{D}_1$ . Moreover, since  $v \in V(X_2) \setminus V(C_2)$ , it follows that  $q \notin \partial\mathcal{D}_2$ . Therefore, there exists a segment of  $\partial\mathcal{D}_2$  that lies inside  $\mathcal{D}_1$ . This implies that  $\varphi_2(X_2)$  intersects the interior of  $\mathcal{D}_1$ . Thus, there exists  $e \notin E(X_1)$ , with  $\varphi_2(e) = \varphi_1(e) \subset \mathcal{D}_1$ , which contradicts the fact that  $(X_1, C_1)$  is a  $\varphi_1$ -patch.  $\square$

We are now ready to prove Lemma 4.4, which is the main result of this section. Before proceeding, we remark that the assertion of Lemma 4.4 can in fact be slightly strengthened. More precisely, one can show that in the computed collection, all patches are universal. We omit the details since they are not relevant to our algorithmic application.

*Proof of Lemma 4.4.* Fix a drawing  $\varphi$  of  $G$  into a surface of Euler genus  $g$ . We remark that we use  $g$  in the following argument, even though we do not know how to explicitly compute it in polynomial time.

We inductively compute a sequence  $\mathcal{P}^{(0)}, \dots, \mathcal{P}^{(s)}$ , with  $\mathcal{P}^{(0)} = \emptyset$ , where for each  $i \in \{1, \dots, s\}$ ,  $\mathcal{P}^{(i)}$  is a collection of pairwise non-overlapping  $\varphi$ -patches of  $G$ . The desired collection will be  $\mathcal{P}^{(s)}$ .

For any  $\ell \in \{1, \dots, s\}$ , we define the graph  $G^{(\ell)} = G \setminus \bigcup_{(X, C) \in \mathcal{P}^{(\ell)}} (X \setminus C)$ . We maintain the inductive invariant that for any  $\ell \in \{1, \dots, s\}$ ,

$$G^{(\ell)} \text{ is normalized, and } \text{eg}(G^{(\ell)}) = \text{eg}(G) = g. \quad (1)$$

Given  $\mathcal{P}^{(\ell)}$ , for some  $\ell \geq 0$ , we proceed as follows. Let  $\alpha > 0$  be the constant in the statement of Lemma 5.2. By Theorem 2.3, there exists a polynomial time algorithm that given a graph of treewidth  $k$ , outputs a tree decomposition of width at most  $\alpha' k \sqrt{\log k}$ , for some universal constant  $\alpha'$ . We run the algorithm of Theorem 2.3 on  $G^{(\ell)}$ . If the algorithm returns a tree decomposition of width at most  $\alpha \cdot \alpha' \cdot \Delta^2 g^6 \log^5 n$ , then we have a certificate that the treewidth of  $G^{(\ell)}$  is at most  $O(\Delta^2 g^6 \log^5 n)$ , and we set  $r = \ell$ . Otherwise, we know that the treewidth of  $G^{(\ell)}$  is at least  $\alpha \Delta^2 g^6 \log^{9/2} n$ , and we proceed to compute  $\mathcal{P}^{(\ell+1)}$ .

By Lemma 5.2, and since  $G^{(\ell)}$  is normalized, we can compute in polynomial time a universal patch in  $(X^*, C^*)$  of  $G^{(\ell)}$ . Let

$$\mathcal{Q}^{(\ell)} = \{(X, C) \in \mathcal{P}^{(\ell)} : (X, C) \text{ and } (X^*, C^*) \text{ are overlapping}\}.$$

Fix an ordering of the patches in  $\mathcal{Q}^{(\ell)} = \{(Y_i, F_i)\}_{i=1}^{k_\ell}$ , where  $k_\ell = |\mathcal{Q}^{(\ell)}|$ . Let  $Y = \bigcup_{i=1}^{k_\ell} Y_i$ . We argue that  $(X^* \cup Y, C)$  is a  $\varphi$ -patch of  $G$ . Let  $\Gamma^{(0)} = G^{(\ell)}$ , and for any  $j \in \{1, \dots, k_\ell\}$ , let  $\Gamma^{(j)} = \Gamma^{(j-1)} \cup Y_j$ . Let also  $\varphi^{(j)}$  be the drawing of  $\Gamma^{(j)}$  induced by restricting  $\varphi$  to  $\Gamma^{(j)}$ . Since

$(X^*, C^*)$  is a universal patch of  $\Gamma^{(0)} = G^{(\ell)}$ , and  $\varphi^{(0)}$  is a drawing into a surface of Euler genus  $g$ , it follows that  $(X^*, C^*)$  is also a  $\varphi^{(0)}$ -patch. Note that for any  $j \in \{1, \dots, k_\ell\}$ , since  $(Y_j, F_j)$  is a  $\varphi$ -patch, and  $\varphi^{(j)}$  is a restriction of  $\varphi$ , it follows that  $(Y_j, F_j)$  is also a  $\varphi^{(j)}$ -patch. By Lemma 5.3 we obtain that  $(X^* \cup Y_1, C^*)$  is a  $\varphi^{(1)}$ -patch of  $\Gamma^{(1)}$ . By inductively applying Lemma 5.3 on the pair of patches  $(Y_j, F_j)$ , and  $(X^* \cup (\bigcup_{i=1}^{j-1} Y_i), C^*)$ , we conclude that  $(X^* \cup Y, C)$  is a  $\varphi^{(k_\ell)}$ -patch of  $\Gamma^{(k_\ell)}$ . Since the patch  $(X^* \cup Y, C)$  is non-overlapping with any of the patches in  $\mathcal{P}^{(\ell)} \setminus \mathcal{Q}^{(\ell)}$ , it follows that  $(X^* \cup Y, C)$  is a  $\varphi$ -patch. We set

$$\mathcal{P}^{(\ell+1)} = \left( \mathcal{P}^{(\ell)} \setminus \mathcal{Q}^{(\ell)} \right) \cup \{(X^* \cup Y, C^*)\}.$$

It is immediate that  $\mathcal{P}^{(\ell+1)}$  is a collection of pairwise non-overlapping  $\varphi$ -patches.

We next show that the inductive invariant (1) is maintained. Observe that

$$G^{(\ell+1)} = G^{(\ell)} \setminus (X^* \setminus C^*).$$

Since  $(X^*, C^*)$  is a universal patch of  $G^{(\ell)}$ , it follows that  $\text{eg}(G^{(\ell+1)}) = \text{eg}(G^{(\ell)}) = g$ . Moreover, by Lemma 5.2 we have that  $G^{(\ell+1)}$  is normalized. This shows that the inductive invariant (1) is maintained.

It remains to argue that the above process terminates after polynomially many steps. Note that since  $(X^*, C^*)$  is a patch, we have  $X^* \subsetneq C^*$ . Therefore,  $G^{(\ell+1)} \subsetneq G^{(\ell)}$ . It follows that the algorithm terminates in polynomial time. This concludes the proof.  $\square$

## 6 Computing a flat grid minor

The last ingredient required for our algorithm is a procedure for computing a universal patch (the proof of Lemma 5.2). This procedure requires a polynomial-time algorithm which given a graph  $G$  of large treewidth, and small Euler genus, computes a large *flat* grid minor in  $G$ . Intuitively, we say that a subgraph  $H$  is flat if it is planar, and moreover it admits a planar drawing, such that all edges leaving  $H$ , are incident to its outer face. A formal definition can be found later in this Section. We give such an algorithm for computing flat grid minors in the present Section. The proof of Lemma 5.2 appears in the following Section.

We remark that all previously known algorithm for computing flat grid minors either work when  $G$  is of constant treewidth, or require a drawing of  $G$  into a small-genus surface as part of the input. In contrast, we need to allow the treewidth to be as large as  $\Omega(n)$ , and we of course do not have access to a low-genus drawing of  $G$  – after all, computing such a drawing is precisely our goal!

Let us now give a high-level overview of our algorithm for finding a flat grid minor. It is known that any graph  $G$  of treewidth  $t$ , and genus  $g$ , contains a  $(\Omega(t/g) \times \Omega(t/g))$ -grid minor. By repeatedly removing balanced vertex-separators, we can compute in polynomial time a set  $X$  of at most  $O(tg \log^{O(1)} n)$  vertices, whose removal leaves a planar graph. In particular, this means that the resulting planar graph  $G \setminus X$  must still contain a relatively large grid minor. It is already known how to compute a large grid minor in a planar graph in polynomial time. Unfortunately, such a grid minor is not guaranteed to be flat in the original graph  $G$ . We argue that a large *subgrid* of the computed grid minor must be flat  $G$ . If this is not the case, then we can find a large apex minor, which contradicts the fact that  $\text{eg}(G) \leq g$ .

We now proceed with the formal proof. The first step towards computing a flat grid minor, is a procedure for computing a small edge set, whose removal leaves a planar graph. This is described in the following lemma.

**Lemma 6.1** (Computing a small planarizing set). *There exists a polynomial-time algorithm which given a graph  $G$  of treewidth  $t$ , and maximum degree  $\Delta$ , and an integer  $g \geq 0$ , either correctly decides that  $\text{eg}(G) > g$ , or it outputs a set  $X \subseteq V(G)$ , satisfying the following conditions:*

- (1)  $|X| = O(gt \log^{5/2} n)$ .
- (2)  $G \setminus X$  is planar.
- (3)  $G \setminus X$  has at most  $O(\Delta g \log n)$  connected components.

*Proof.* We inductively define a collection  $\mathcal{F}$  of subgraphs of  $G$ , and an auxiliary tree  $T$  with  $V(T) = \mathcal{F}$ . Initially, we set  $\mathcal{F}_0 = \{G\}$ , and we let  $T_0$  be the tree with single vertex  $G$ , which we also consider to be its root. Given  $\mathcal{F}_i$ , and  $T_i$ , we proceed as follows. If all leaves of  $T_i$  are planar subgraphs of  $G$ , then we set  $\mathcal{F} = \mathcal{F}_i$ , and  $T = T_i$ , concluding the construction of  $\mathcal{F}$ , and  $T$ . Otherwise, let  $\mathcal{L}_i$  be the set of leaves of  $T$  that are non-planar subgraphs of  $G$ . Each  $H \in \mathcal{L}_i$  is a subgraph of  $G$ , and therefore has treewidth at most  $t$ . Therefore,  $H$  contains a  $\Theta(1)$ -balanced vertex separator  $S_H$ , with  $|S_H| \leq t$ . Using the algorithm from Theorem 2.3, we can compute a  $\Theta(1)$ -balanced vertex separator  $S'_H$  of  $H$ , with  $|S'_H| = O(|S_H| \sqrt{\log n}) = O(t \sqrt{\log n})$ . Moreover, we can ensure that  $H \setminus S'_H$  has at most  $\Delta$  connected components (this can be done by taking a minimal subset of  $S'_H$  that forms a  $\Theta(1)$ -balanced vertex separator of  $H$ ). Let  $\mathcal{C}_H$  be the set of connected components of  $H \setminus S'_H$ . We set

$$\mathcal{F}_{i+1} = \mathcal{F}_i \cup \bigcup_{H \in \mathcal{L}_i} \mathcal{C}_H$$

We also construct  $T_{i+1}$  as follows. Initially, we set  $T_{i+1} = T_i$ . For every  $C \in \mathcal{C}_H$ , we add the vertex  $C$  in  $V(T_{i+1})$ , and we add the edge  $\{H, C\} \in E(T)$ . Note that  $H$  is the parent of  $C$  in the rooted tree  $T_{i+1}$ . We continue until all the leaves of the current tree are planar subgraphs of  $G$ . This completes the definition of  $\mathcal{F}$ , and  $T$ .

Let  $\mathcal{L}$  be the set of leaves of  $T$ . We can now define

$$X = \bigcup_{H \in \mathcal{F} \setminus \mathcal{L}} S'(H).$$

It is immediate by the construction that  $G \setminus X = \bigcup_{H \in \mathcal{L}} H$ . Since all leaves of  $T$  are planar subgraphs of  $G$ , it follows that every connected component of  $G \setminus X$  is planar, and therefore  $G \setminus X$  is planar.

It remains to bound  $|X|$ . Observe that for any  $i > 0$ , every graph in  $\mathcal{F}_i$  is a connected component of  $H \setminus S'_H$ , for some  $H \in \mathcal{F}_{i-1}$ . Since  $S'_H$  is a  $\Theta(1)$ -balanced separator of  $H$ , it follows that the maximum size of every graph in  $\mathcal{F}_i$  is at most  $2^{-ic}$ , for some constant  $c > 0$ . Therefore, the depth of  $T$  is  $h = O(\log n)$ .

For every  $i \in \{0, \dots, h\}$ , let  $\mathcal{Z}_i$  be the set of all graphs  $H \in \mathcal{F}$ , such that  $H$  is at depth  $i$  in  $T$ , and it is non-planar. For any  $i \in \{0, \dots, h\}$ , we have

$$\bigcup_{H \in \mathcal{Z}_i} H \subseteq G.$$

Since all graphs in  $\mathcal{Z}_i$  are pairwise vertex-disjoint subgraphs of  $G$ , we have

$$\sum_{H \in \mathcal{Z}_i} \text{eg}(H) \leq \text{eg}(G) = g.$$

Since all graphs in  $\mathcal{Z}_i$  are non-planar, we have that for any  $H \in \mathcal{Z}_i$ ,  $\text{eg}(H) \geq 1$ . Therefore,

$$|\mathcal{Z}_i| \leq g.$$

We have

$$\begin{aligned} |X| &= \left| \bigcup_{H \in \mathcal{F} \setminus \mathcal{L}} S'(H) \right| \\ &= \left| \bigcup_{i=0}^h \bigcup_{H \in \mathcal{Z}_i} S'(H) \right| \\ &\leq O(hgt \log^{3/2} n) \\ &\leq O(gt \log^{5/2} n), \end{aligned}$$

as required. By construction, every separator we compute increases the number of connected components by at most  $\Delta - 1$ . It follows that the total number of connected components of  $G \setminus X$  is at most  $O(\Delta g \log n)$ .  $\square$

The following lemma gives a trade-off between the treewidth of a genus  $g$  graph, and the size of its largest grid minor.

**Lemma 6.2** (Demaine et al. [DFHT05]). *Let  $G$  be a graph of Euler genus  $g$ , and treewidth  $t$ . Then,  $G$  contains a  $(r \times r)$ -grid minor, for some  $r \geq \frac{t+1}{6(g+1)}$ .*

The following lemma shows that when deleting a small subset of a graph, a moderately large grid minor “survives”. This property will be used crucially by our algorithm for computing a grid minor. An earlier version of the present paper gave a slightly weaker bound on the size of the resulting grid minor. Our bound has been subsequently improved by Eppstein [Epp13], who obtained an asymptotically optimal estimate.

**Lemma 6.3** (Eppstein [Epp13]). *Let  $r, f \geq 1$ . Let  $G$  be the  $(r \times r)$ -grid, and  $X \subset V(G)$ , with  $|X| = f$ . Then,  $G \setminus X$  contains the  $(r' \times r')$ -grid as a minor, where  $r' = \max\{r - f, r^2/4f - O(1)\} = \Theta(\min\{r, r^2/f\})$ .*

The following lemma is an intermediate step towards getting a flat grid minor. The main missing property is that the grid minor guaranteed by Lemma 6.4 might not be flat. We will subsequently ensure flatness via a more elaborate argument.

**Lemma 6.4.** *Let  $G$  be a graph of Euler genus  $g \geq 1$ , treewidth  $t \geq 1$ , and maximum degree  $\Delta$ . Then, in polynomial time we can compute a planar subgraph  $\Gamma \subseteq G$  satisfying the following conditions:*

- (1) *The graph  $\Gamma$  contains the  $(r' \times r')$ -grid as a minor, with  $r' = \Omega\left(\frac{t}{g^3 \log^{5/2} n}\right)$ .*

(2) The graph  $G \setminus \Gamma$  has at most  $O(\Delta g \log n)$  connected components.

We remark that we do not require a drawing of  $G$  as part of the input.

*Proof.* By lemma 6.2, we have that  $G$  contains as a minor the  $(r \times r)$ -grid  $H$ , for some  $r \geq \frac{t+1}{6(g+1)} > \frac{t}{16g}$ . More precisely, for every  $v \in V(H)$ , there exists  $U_v \subseteq V(G)$ , such that the following conditions are satisfied:

- For any  $v, v' \in V(H)$ ,  $U_v \cap U_{v'} = \emptyset$ .
- For any  $\{v, v'\} \in E(H)$ , there exist  $u \in U_v$ , and  $u' \in U_{v'}$ , such that  $\{u, u'\} \in E(G)$ .

Using the algorithm from Lemma 6.1, we compute a set  $X \subseteq V(G)$ , with  $|X| = O(gt \log^{5/2} n)$ , such that the graph  $G \setminus X$  is planar. We first argue that  $G \setminus X$  contains a large grid minor. Let

$$X' = \{v \in V(H) : X \cap U_v \neq \emptyset\}.$$

By Lemma 6.3, we have that  $H \setminus X'$  contains as a subgraph a  $(r' \times r')$ -grid  $H'$ , for some

$$\begin{aligned} r' &= \Omega(\min\{r, r^2/f\}) \\ &= \Omega\left(\min\left\{\frac{t}{g}, \frac{t^2}{g^3 t \log^{5/2} n}\right\}\right) \\ &= \Omega\left(\frac{t}{g^3 \log^{5/2} n}\right). \end{aligned}$$

Since  $H \setminus X'$  is a minor of  $G \setminus X$ , and  $H'$  is a subgraph of  $H$ , it follows that  $H'$  is a minor of  $G \setminus X$ . By setting  $\Gamma$  to be the connected component of  $G \setminus X$  containing  $H'$  as a minor, assertion (1) follows. To prove assertion (2), it suffices to note that  $G \setminus X$  has at most  $c = O(\Delta g \log n)$  connected components, which implies that  $G \setminus \Gamma$  also has at most  $c - 1 = O(\Delta g \log n)$  connected components.  $\square$

The following result gives a polynomial-time constant-factor approximation for computing a maximum grid minor in a planar graph. It is obtained by combining the algorithm for computing a branch decomposition due to Seymour, and Thomas [ST94], with the grid minor algorithm of Robertson, Seymour, and Thomas [RST94] (see also [CKS04, GT11]).

**Lemma 6.5** (Seymour, and Thomas [ST94], Robertson, Seymour, and Thomas [RST94]). *Let  $r > 0$ , and let  $G$  be a planar graph containing a  $(r \times r)$ -grid minor. Then, on input  $G$ , we can compute in polynomial time a  $(\Omega(r) \times \Omega(r))$ -grid minor in  $G$ .*

We now formally define the notion of *flatness*.

**Definition 6.6** (Flatness). *Let  $G$  be a graph, and let  $H \subseteq G$  be a planar subgraph. We say that  $H$  is flat (w.r.to  $G$ ) if there exists a planar drawing of  $H$ , such that for all edges  $\{u, v\} \in E(G)$ , with  $u \in V(H)$ , and  $v \in V(G) \setminus V(H)$ , the vertex  $u$  is in the outer face of  $H$ .*

Our strategy for computing a flat grid minor is as follows. We first remove a small set of edges from the input graph  $G$ , so that the resulting graph is planar  $\Gamma$ . A key property is that  $\Gamma$  consists of a small number of connected components, and has large treewidth. This means that it also has a large grid minor. We argue that some sub-grid of this grid-minor must be flat; otherwise, we can find a large apex minor in  $G$ , which contradicts the fact that the Euler genus is small. The following technical Lemma will be used later in our proof when finding the desired apex minor.

**Lemma 6.7.** *Let  $l, k \geq 1$  be integers. Let  $f : \{1, \dots, l\}^2 \rightarrow \{1, \dots, k\}$  be a coloring of the  $l \times l$  grid with  $k$  colors. Then, there exists a color  $c^* \in \{1, \dots, k\}$ , and a set of rows  $I \subseteq \{1, \dots, l\}$ , with  $|I| \geq l/k$ , such that every row in  $I$  has at least  $l/k$  elements with color  $c^*$ .*

*Proof.* For row  $i \in \{1, \dots, l\}$ , there exists at least one color  $c_i \in \{1, \dots, k\}$ , such that row  $i$  contains at least  $l/k$  elements with color  $c_i$ . Since there are  $k$  colors, it follows that at least one color  $c$  is a most frequent color in at least  $l/k$  rows. The assertion follows.  $\square$

For completeness, we also give a proof of the following easy fact about the Euler genus of apex graphs. A more complete treatment of the (orientable) genus of apex graphs can be found in [Moh01].

**Lemma 6.8.** *Let  $k \geq 3$ , and let  $H$  be the  $(k \times k)$ -grid. Let  $G$  be the apex graph obtained from  $H$  by adding a new vertex  $v^*$ , and connecting it to every vertex in  $V(H)$ . Then,  $\text{eg}(G) = \Omega(k^2)$ .*

*Proof.* The graph  $G$  contains  $\Omega(k^2)$  edge-disjoint subgraphs, each of Euler genus 1, and such that any two of them share only the vertex  $v^*$ . By additivity properties of genus (e.g. [BHK62]), it follows that  $\text{eg}(G) = \Omega(k^2)$ .  $\square$

We are now ready to give a polynomial-time algorithm for computing a flat grid minor. This is the main result of this Section.

**Lemma 6.9** (Computing a flat grid minor). *There exists a universal constant  $\alpha > 0$ , such that the following holds. There exists a polynomial-time algorithm which given a graph  $G$  of treewidth  $t$ , and maximum degree  $\Delta$ , and an integer  $g \geq 1$ , such that  $t \geq \alpha \Delta^2 g^6 \log^{9/2} n$ , either correctly decides that  $\text{eg}(G) > g$ , or it outputs a flat subgraph  $X \subset G$ , such that  $X$  contains a  $(\Omega(r) \times \Omega(r))$ -grid minor  $M$ , for some  $r = \Omega\left(\frac{t^{1/2}}{g^{3/2} \log^{5/4} n}\right)$ . Moreover, in the latter case, the algorithm also outputs a minor mapping for  $M$ .*

*Proof.* We first use Lemma 6.4 to find a planar subgraph  $\Gamma \subseteq G$ , such that  $\Gamma$  contains a  $(r' \times r')$ -grid minor, for some  $r' = \Omega\left(\frac{t}{g^3 \log^{5/2} n}\right)$ , and  $G \setminus \Gamma$  has at most  $c = O(\Delta g \log n)$  connected components. Using Lemma 6.5 we can compute a  $(k \times k)$ -grid minor  $H$  in  $\Gamma$ , for some  $k = \Omega(r')$ .

For  $i, j \in \{1, \dots, k\}$ , let  $v_{i,j}$  be the vertex in the  $i$ -th row, and  $j$ -th column of  $H$ . For  $i, j \in \{1, \dots, \lfloor k^{1/2} \rfloor\}$ , let

$$U_{i,j} = \bigcup_{i',j' \in \{0, \dots, \lfloor k^{1/2} \rfloor\}} v_{i \cdot \lfloor k^{1/2} \rfloor + i', j \cdot \lfloor k^{1/2} \rfloor + j'}$$

Fix a minor mapping  $\mu : V(H) \rightarrow 2^{V(\Gamma)}$  for  $H$ . Note that we can choose  $\mu$  so that

$$\bigcup_{v \in V(H)} \mu(v) = V(\Gamma).$$

For  $i, j \in \{1, \dots, \lfloor k^{1/2} \rfloor\}$ , let

$$\Gamma_{i,j} = \Gamma[\mu(U_{i,j})].$$

Fix a planar drawing  $\varphi$  of  $\Gamma$ . Suppose first that there exists  $i, j \in \{1, \dots, \lfloor k^{1/2} \rfloor\}$ , such that  $N(\Gamma_{i,j}) \subseteq \Gamma$ , i.e.  $\Gamma_{i,j}$  does not have any neighbors in  $G \setminus \Gamma$ . Then, we claim that  $\Gamma_{i,j}$  is flat w.r.to  $G$ . So see this, let  $\varphi_{i,j}$  be the planar drawing of  $\Gamma_{i,j}$  induced by  $\varphi$ . Recall that for every  $y \in V(\Gamma)$ ,

there exists  $v \in V(H)$ , such that  $y \in \mu(v)$ . Therefore, for every edge  $\{x, y\} \in E(\Gamma_{i,j}, G \setminus \Gamma_{i,j})$ , the vertex  $x$  is in the outer face of  $\varphi_{i,j}$ . It follows that  $\Gamma_{i,j}$  is flat w.r.to  $G$ . Note that  $\Gamma_{i,j}$  contains a  $(h \times h)$ -grid minor, where  $h = \lfloor k^{1/2} \rfloor = \Omega\left(\frac{t^{1/2}}{g^{3/2} \log^{5/4} n}\right)$ , as required.

It remains to consider the case where for every  $i, j \in \{1, \dots, \lfloor k^{1/2} \rfloor\}$ , there exists an edge  $e_{i,j} = \{x_{i,j}, y_{i,j}\}$ , with  $x_{i,j} \in V(\Gamma_{i,j})$ , and  $y_{i,j} \in G \setminus \Gamma$ . Construct a coloring of the  $\lfloor k^{1/2}/2 \rfloor \times \lfloor k^{1/2}/2 \rfloor$  grid  $Z$  by  $c$  colors as follows. Number the connected components of  $G \setminus \Gamma$  by  $1, \dots, c$ . For every  $i, j \in \{1, \dots, \lfloor k^{1/2}/2 \rfloor\}$ , if  $y_{2i,2j}$  is contained in the connected component of  $G \setminus \Gamma$  numbered  $c_{i,j}$ , then we assign to the element in the  $i$ -th row, and  $j$ -th column of  $Z$  the color  $c_{i,j}$ . Let

$$k' = \lfloor k^{1/2}/(2c) \rfloor.$$

By Lemma 6.7, there exists  $c^* \in \{1, \dots, c\}$ , and  $I \subseteq \{1, \dots, \lfloor k^{1/2}/2 \rfloor\}$ , with  $|I| = k'$ , such that for every  $i \in I$ , at least  $k'$  elements in the  $i$ -th row of  $Z$  are assigned the color  $c^*$ . For every  $i \in I$ , pick a set  $J_i \subset \{1, \dots, \lfloor k^{1/2} \rfloor\}$ , with  $|J_i| = k'$ , such that for every  $j \in J_i$ , the element in the  $i$ -th row, and  $j$ -th column of  $Z$  is assigned color  $c^*$ .

Let  $I = \{i_1, \dots, i_{k'}\}$ , and for every  $i \in I$ , let  $J_i = \{j_{i,1}, \dots, j_{i,k'}\}$ . Observe that for any  $s \in \{1, \dots, k'\}$ , there exists a collection of pairwise vertex-disjoint paths  $P_{s,1}, \dots, P_{s,k'}$ , satisfying the following conditions:

- For very  $t \in \{1, \dots, k'\}$ , the endpoints of the path  $P_{s,t}$  are a vertex in  $\Gamma_{2i_s, 2j_{i_s,t}}$ , and a vertex in  $\Gamma_{2i_{s+1}, 2j_{i_{s+1},t}}$ .
- For very  $t \in \{1, \dots, k'\}$ , all the interior vertices in  $P_{s,t}$  are contained in  $\bigcup_{j=1}^{\lfloor k^{1/2} \rfloor} \Gamma_{2i_s+1, j}$ .

Let  $C^*$  be the connected component of  $G \setminus \Gamma$  numbered  $c^*$ . It follows that there exists a  $(k' \times k')$ -grid minor  $H'$  in  $\Gamma$ , with minor mapping  $\mu'$ , such that for every  $v \in V(H')$ , there exists an edge between a vertex in  $\mu'(v)$ , and a vertex in  $C^*$ . Contracting  $C^*$  into a single vertex, we get as a minor the graph obtained from the  $(k' \times k')$ -grid, after adding a new vertex, and connecting it to all vertices. It follows by Lemma 6.8 that  $g = \text{eg}(G) = \Omega((k')^2) = \Omega\left(\frac{t}{\Delta^2 g^5 \log^{9/2} n}\right)$ , and thus  $t = O(\Delta^2 g^6 \log^{9/2} n)$ , which is a contradiction for a sufficiently large constant  $\alpha > 0$ . This concludes the proof.  $\square$

## 7 Computing a universal patch

In this Section, we present the last missing ingredient of our drawing algorithm: a procedure for computing a universal patch. Our algorithm for computing universal patches uses as a subroutine the procedure for computing a flat grid minor from Section 6.

Our proof uses powerful machinery developed by Mohar [Moh92]. The tools from [Moh92] allow us to guarantee that certain cycles of a given graph are contractible in *any* optimal drawing. This property will be crucial when computing a universal patch. We begin with some definitions.

Let  $H$  be a subgraph of a graph  $G$ . An  $H$ -component of  $G$  is either an edge in  $E(G)$  with both endpoints in  $V(H)$ , or it is a connected component  $X$  of  $G \setminus V(H)$  together with all the edges between  $X$  and  $H$ . Each edge of an  $H$ -component  $Y$  with an endpoint in  $V(H)$  is a *foot* of  $Y$ .

**Definition 7.1** (Planarly nested sequence [Moh92]). *A sequence  $C_1, \dots, C_k$  of disjoint cycles in a graph  $G$  is planarly nested if for any  $i \in \{1, \dots, k\}$ , there exists a  $C_i$ -component  $H_i$ , such that  $H_1 \supset \dots \supset H_k$ , and the graph obtained from  $G$  by contracting to a single vertex all edges in the  $C_i$ -component  $H_i$ , except its feet, is planar.*

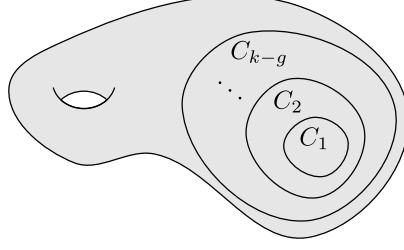


Figure 5: Planarly nested sequence for Lemma 7.2

The following Lemma is due to Mohar [Moh92]. In fact, [Moh92] derives a somewhat more precise bound for non-orientable drawings. We will use a slightly weaker form which simplifies the notation, but is still sufficient for our application.

**Lemma 7.2** (Mohar [Moh92]). *Let  $G$  be a graph of Euler genus  $g$ . Let  $\varphi$  be a drawing of  $G$  into a surface  $S$  of Euler genus  $g$ . Let  $C_1, \dots, C_k$  be a planarly nested sequence of cycles in  $G$ , where  $k > g$ . Then, the cycles  $\varphi(C_1), \dots, \varphi(C_{k-g})$  bound disks in  $S$  (see Figure 5).*

The following crucial fact about planarly nested sequences, allows us to inductively maintain a graph of Euler genus  $g$ .

**Lemma 7.3** (Mohar [Moh92]). *Let  $G$  be a graph of Euler genus  $g$ , and let  $C_1, \dots, C_k$  be a planarly nested sequence in  $G$ , with  $k > g+1$ . Let  $H$  be the  $C_1$ -component containing  $C_2$ , and let  $G' = C_1 \cup H$ . Then,  $\text{eg}(G') = \text{eg}(G) = g$ .*

We can use our algorithm for computing a flat grid minor (lemma 6.9), to compute a planarly nested sequence. The next lemma gives the precise statement.

**Lemma 7.4** (Computing a planarly nested sequence). *There exists a universal constant  $\alpha > 0$ , such that the following holds. There exists a polynomial-time algorithm which given a graph  $G$  of treewidth  $t$ , and maximum degree  $\Delta$ , and an integer  $g \geq 1$ , with  $t \geq \alpha \Delta^2 g^6 \log^{9/2} n$ , either correctly decides that  $\text{eg}(G) > g$ , or it outputs a planarly nested sequence  $C_1, \dots, C_k$  in  $G$ , with  $k = \Omega\left(\frac{t^{1/2}}{g^{3/2} \log^{5/4} n}\right)$ .*

*Proof.* By Lemma 6.9, we can compute in polynomial time a flat subgraph  $H \subset G$ , containing a  $(r \times r)$ -grid minor  $M$ , for some  $r = \Omega\left(\frac{t^{1/2}}{g^{3/2} \log^{5/4} n}\right)$ , together with a minor mapping  $f$  for  $M$ .

Since  $H$  is flat, we can compute a planar drawing  $\varphi$  of  $H$ , such that all the edges in  $E(V(H), V(G) \setminus V(H))$  are incident to the unbounded face of  $\varphi$ . The drawing  $\varphi$  induces a planar drawing  $\varphi'$  of  $M$ . There exist a sequence of at least  $k = \lfloor r/4 \rfloor$  pairwise disjoint cycles  $C'_1, \dots, C'_k$  in  $M$ , such that for any  $i \in \{1, \dots, k\}$ , the cycle  $\varphi'(C'_i)$  bounds a disk  $\mathcal{D}'_i$  in the plane, for any  $j \in \{1, \dots, k-1\}$ ,  $\varphi'(C'_j) \subset \mathcal{D}'_{j+1}$ , and there exists  $x \in V(G)$ , such that  $\varphi'(x)$  lies in the interior of  $\mathcal{D}'_1$ . For each  $i \in \{1, \dots, k\}$ , we can find a cycle  $C_i$  in  $H$ , such that  $V(C_i) \subseteq f(V(C'_i))$ . It is straightforward to check that the sequence of cycles  $C_1, \dots, C_k$  is planarly nested.  $\square$

The next lemma gives a technical condition that follows by the properties of normalized graphs, and will be used by our algorithm for computing a universal path.



**Lemma 7.5** (Uniqueness of the interior). *Let  $G$  be a connected normalized graph, and let  $C, F \subseteq G$  be vertex-disjoint cycles. Let  $\varphi, \varphi'$  be drawings of  $G$  into surfaces  $\mathcal{S}$ , and  $\mathcal{S}'$  respectively. Suppose that there exist disks  $\mathcal{D}_C, \mathcal{D}_F \subset \mathcal{S}$ , and  $\mathcal{D}'_C, \mathcal{D}'_F \subset \mathcal{S}'$ , with  $\mathcal{D}_C \subset \mathcal{D}_F$ , and  $\mathcal{D}'_C \subset \mathcal{D}'_F$ , such that  $\partial \mathcal{D}_C = \varphi(C)$ ,  $\partial \mathcal{D}_F = \varphi(F)$ ,  $\partial \mathcal{D}'_C = \varphi'(C)$ , and  $\partial \mathcal{D}'_F = \varphi'(F)$ . Then, the set of vertices (resp. edges) that lie inside  $\mathcal{D}_C$  in the drawing  $\varphi$ , is the same as the set of vertices (resp. edges) that lie inside  $\mathcal{D}'_C$  in the drawing  $\varphi'$ , i.e.  $\varphi^{-1}(\mathcal{D}_C \cap \varphi(G)) = \varphi'^{-1}(\mathcal{D}'_C \cap \varphi'(G))$ .*

*Proof.* Let

$$X = \varphi^{-1}(\mathcal{D}_C \cap \varphi(G)),$$

and

$$X' = \varphi'^{-1}(\mathcal{D}'_C \cap \varphi'(G)).$$

We need to show that  $X = X'$ . Let  $H$  be the  $F$ -component of  $G$  containing  $C$ . Since  $G$  is connected, and  $C$  separates both  $X$ , and  $X'$  from  $F$ , we have  $X \subseteq H$ , and  $X' \subseteq H$ . Let  $G' = H \cup F$ . Let also  $G''$  be the graph obtained from  $G'$  by adding a new vertex  $f$ , and connecting it to every vertex in  $V(F)$ . Clearly,  $G''$  is a planar graph.

Let  $\Gamma''$  be the graph obtained from  $G''$  by replacing all maximal induced paths by edges. We argue that  $\Gamma''$  is 3-connected. For the sake of contradiction suppose that there exists a separator  $Z \subset V(\Gamma'')$ , with  $|Z| \leq 2$ . Observe that if  $f \notin Z$ , then some  $Z$ -component of  $\Gamma''$  induces a free subgraph in  $G''$ , which much also be a free in  $G$ , which contradicts the fact that  $G$  is normalized. Therefore, we may assume that  $f \in Z$ . Let  $Z = \{f, f'\}$ . Since  $f$  is connected to every vertex in  $V(F)$ , and  $F$  is a connected subgraph, it follows that there exists a component of  $\Gamma'' \setminus Z$  that contains  $F$ . This implies however that  $\{f'\}$  is also a separator in  $\Gamma''$ . It follows that some  $\{f'\}$ -component must be free in  $\Gamma''$ , and therefore also in  $G$ , which again contradicts the fact that  $G$  is normalized. We have thus established that  $G''$  is a subdivision of a 3-connected graph. The restrictions of  $\varphi$ , and  $\varphi'$  on  $\mathcal{D}_F$ , and  $\mathcal{D}'_F$  respectively, can be extended to drawings  $\psi$ , and  $\psi'$  of  $G''$ . Since  $G''$  is the subdivision of a planar graph, it follows that it admits a combinatorially unique planar drawing. This implies that  $X = X'$ , which concludes the proof.  $\square$

We are now ready to give our algorithm for computing a universal patch. This is the main result of this section.

*Proof of Lemma 5.2.* By Lemma 7.4 there exists a universal constant  $\beta$ , such that we can compute a planarly nested sequence  $C_1, \dots, C_k$  in  $G$ , for some  $k \geq \beta \frac{t^{1/2}}{g^{3/2} \log^{5/4} n}$ . For a sufficiently large constant  $\alpha > 0$ , we get  $k \geq g + 3$ . Let  $\varphi$  be a drawing of  $G$  into a surface  $\mathcal{S}$  of Euler genus  $g$ . By Lemma 7.2 we have that the cycles  $\varphi(C_1), \varphi(C_2), \varphi(C_3)$  bounds disks  $\mathcal{D}_1, \mathcal{D}_2, \mathcal{D}_3$  respectively, with

$$\mathcal{D}_1 \subset \mathcal{D}_2 \subset \mathcal{D}_3.$$

We now define a sequence of cycles  $\Psi^{(0)}, \dots, \Psi^{(r)}$ , where  $\Psi^{(0)} = C_2$ , and  $\Psi^{(r)}$  will be the desired cycle  $C$ . For every  $i \in \{1, \dots, r\}$ , let  $H^{(i)}$  be the connected component of  $G \setminus \Psi^{(i)}$  containing  $C_3$ . Let also  $W^{(i)} = \Psi^{(i)} \cup H^{(i)}$ , and  $X^{(i)} = G \setminus W^{(i)}$ . Suppose we are given  $\Psi^{(i)}$ . If the graph  $W^{(i)}$  is normalized, then we set  $r = i$ , and therefore  $C = \Psi^{(i)}$ . Otherwise, we proceed to define  $\Psi^{(i+1)}$ . Since  $W^{(i)}$  is not normalized, this means that there exists a free vertex-induced subgraph  $Q^{(i)}$  of  $W^{(i)}$ . Since  $G$  is normalized, it follows that  $Q^{(i)}$  is not free in  $G$ . Therefore,  $V(Q^{(i)}) \cap V(\Psi^{(i)}) \neq \emptyset$ .

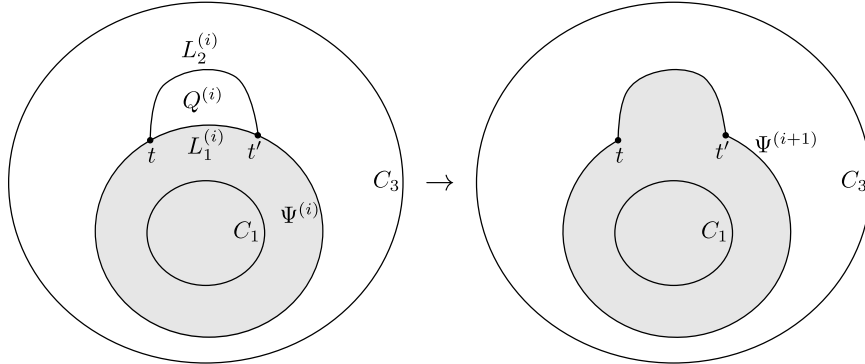
We first argue that  $Q^{(i)}$  cannot be a petal in  $W^{(i)}$ . Suppose that, to the contrary,  $Q^{(i)}$  is a petal, with portal some vertex  $t$ . If  $t \in V(\Psi^{(i)})$ , then all edges between  $V(Q^{(i)})$ , and  $V(G \setminus Q^{(i)})$

in  $G$ , must be incident to  $t$ . This implies that  $Q^{(i)}$  is also a vertex-induced subgraph of  $G$ , and therefore also a petal in  $G$ , which contradicts the fact that  $G$  is normalized. Therefore, we must have  $t \in V(H^{(i)})$ . If  $V(Q^{(i)}) \cap V(\Psi^{(i)}) = \emptyset$ , then clearly  $Q^{(i)}$  is a petal in  $G$ , which, again, is a contradiction. Thus, it must be  $V(Q^{(i)}) \cap V(\Psi^{(i)}) \neq \emptyset$ . Since  $\Psi^{(i)}$  is a 2-connected subgraph, and  $Q^{(i)}$  is a petal in  $W^{(i)}$ , it follows that  $\Psi^{(i)} \subseteq Q^{(i)}$ . Observe that  $\Psi^{(i)}$  bounds a disk in  $\varphi$ , and therefore separates  $X^{(i)}$  in  $G$ . This implies that  $Q^{(i)} \cup X^{(i)}$  is a petal in  $G$ , which again contradicts the fact that  $G$  is normalized. This completes the proof that  $Q^{(i)}$  cannot be a petal in  $W^{(i)}$ .

Therefore,  $Q^{(i)}$  is a clump in  $W^{(i)}$ . Let  $t, t'$  be the portals of  $Q^{(i)}$ . If none of the portals are in  $V(\Psi^{(i)})$ , then by arguing as above we deduce that  $Q^{(i)}$  must be a clump in  $G$ , which contradicts the fact that  $G$  is normalized. Thus, we may assume  $t \in V(\Psi^{(i)})$ . If  $t' \notin V(\Psi^{(i)})$ , then it must be  $t' \in V(H^{(i)})$ . But in this case, we again conclude that  $Q^{(i)}$  must be a clump in  $G$ , which is a contradiction. Therefore, we must have  $t, t' \in V(\Psi^{(i)})$ .

We argue that  $V(Q^{(i)}) \cap V(C_3) = \emptyset$ . Suppose to the contrary that  $V(Q^{(i)}) \cap V(C_3) \neq \emptyset$ . Since both portals of  $Q^{(i)}$  are in  $V(\Psi^{(i)})$ , it follows that  $H^{(i)} = Q^{(i)}$ , which contradicts the fact that  $Q^{(i)}$  is planar. This establishes that  $V(Q^{(i)}) \cap V(C_3) = \emptyset$ .

Since  $V(Q^{(i)}) \cap V(C_3) = \emptyset$ , this means that  $\varphi(Q^{(i)}) \subset \mathcal{D}_3$ . Therefore, the drawing  $\varphi$ , induces a planar drawing  $\psi^{(i)}$  of  $Q^{(i)}$ . The outer face of  $\psi^{(i)}$  consists of two walks  $L_1^{(i)}$ , and  $L_2^{(i)}$  between  $t$ , and  $t'$ . Let  $K_1^{(i)}, K_2^{(i)}$  be the two paths in  $\Psi^{(i)}$  between  $t$ , and  $t'$ . There exists  $l \in \{1, 2\}$ ,  $k \in \{1, 2\}$ , such that the cycle  $\varphi(L_l^{(i)} \cup K_k^{(i)})$  bounds a disk  $\mathcal{F}^{(i)} \subset \mathcal{S}$ , with  $\varphi(L_{1-l}^{(i)} \cup K_{1-k}^{(i)}) \subset \mathcal{F}^{(i)}$ . We can now define  $\Psi^{(i+1)} = L_l^{(i)} \cup K_k^{(i)}$ .



This completes the definition of the sequence  $\Psi^{(0)}, \dots, \Psi^{(t)}$ .

We first show that the sequence terminates at a finite  $t \geq 0$ . Assume  $t > 0$ , since otherwise there is nothing to show. For every  $i \in \{0, \dots, t-1\}$ , we have  $X^{(i+1)} \supseteq X^{(i)}$ . Moreover, since  $Q^{(i)}$  is free, it cannot consist of a single path between its terminals. Therefore, there exists at least one vertex  $x \in L_{1-l}^{(i)} \setminus L_l^{(i)}$ . Therefore,  $x \in X^{(i+1)}$ , and  $x \notin X^{(i)}$ , which implies that  $X^{(i+1)} \supsetneq X^{(i)}$ . This implies that the sequence  $\Psi^{(0)}, \Psi^{(1)}, \dots$ , terminates at some  $\Psi^{(t)}$ , for some finite  $t \geq 0$ .

We can now define  $X = X^{(t)} \cup \Psi^{(t)}$ . We have already shown that  $G \setminus (X \setminus C)$  is normalized. It remains to show that  $(X, C)$  is a universal patch in  $G$ . Let  $\varphi'$  be a drawing of  $G$  into a surface  $\mathcal{S}'$  of Euler genus  $g$ . The sequence of cycles  $C_1, C, C_3, \dots, C_k$  is planarly nested, and therefore by Lemma 7.2 we have that the cycle  $\varphi'(C)$  bounds a disk  $\mathcal{D}'_C \subset \mathcal{S}'$ . Similarly, the cycle  $\varphi'(C_3)$  bounds a disk  $\mathcal{D}_F \subset \mathcal{S}'$ . By Lemma 7.5, setting  $F = C_3$ , we have that in any such drawing  $\varphi'$ , the sets of vertices, and edges of  $G$  that are mapped inside  $\mathcal{D}'_C$ , are uniquely determined. This proves condition (1) of Definition 5.1. Condition (2) follows directly by applying Lemma 7.3 on the planarly nested

sequence  $C, C_3, \dots, C_k$ . Therefore,  $(X, C)$  is a universal patch, as required. This concludes the proof.  $\square$

## 8 Approximating the orientable genus

In this section we give an algorithm for approximating the *orientable* genus of a graph. The main idea is the following. Let  $G$  be the given graph of maximum degree  $\Delta$ , and let  $g = \text{genus}(G)$ . We first use the algorithm from Theorem 1.1 to compute a drawing of  $G$  into a surface of Euler genus  $g' = \Delta^{O(1)} g^{O(1)} \log^{O(1)} n$ . If the resulting surface is orientable, then we are done.<sup>2</sup> Otherwise, we argue that there exists a set of  $g^{O(1)}$  vertices, whose removal decreases the genus of the current surface. More precisely, we show that if this is not the case, then there exists a large “Möbius” grid minor, i.e. a graph “densely” embedded into the Möbius band (to be formally defined later). Such a minor has large orientable genus, which leads to a contradiction. After repeating this procedure at most  $O(g')$  times, we arrive at a drawing of a subgraph  $G' \subseteq G$ , into some orientable surface of genus at most  $g'$ . We extend this drawing to a drawing of  $G$ , simply by adding one handle for every removed edge. This results into the desired drawing.

We begin by recalling some standard definitions, capturing the notion of a “dense” embedding. Let  $G$  be a graph, and let  $\varphi$  be a drawing of  $G$  into a surface  $\mathcal{S}$ . A *noose* (in  $\varphi$ ) is a loop in  $\mathcal{S}$ , meeting  $\varphi(G)$  only on  $\varphi(V(G))$ . The *length* of a noose  $\gamma$  is defined to be

$$\text{len}(\gamma) = |\{v \in V(G) : \varphi(v) \in \gamma\}|.$$

The *representativity* of  $\varphi$  is defined to be the smallest length of all noncontractible nooses in  $\varphi$ . In a similar vein, we also say that a curve  $\xi$  with distinct endpoints, is a *chain* (in  $\varphi$ ), if it meets  $\varphi(G)$  only on  $\varphi(V(G))$ . The *length* of a chain  $\xi$  is defined to be

$$\text{len}(\xi) = |\{v \in V(G) : \varphi(v) \in \xi\}| - 1.$$

The following result by Fiedler et al. [FHRR95], gives an obstruction to orientable genus, in terms of the representativity of projective graphs.

**Lemma 8.1** (Fiedler et al. [FHRR95]). *Let  $G$  be a graph drawn into the projective plane, with representativity  $\rho \neq 2$ . Then  $\text{genus}(G) = \lfloor \rho/2 \rfloor$ .*

We now define the graph which we will use as an obstruction to the orientable genus of graphs drawn into a nonorientable surface.

**Definition 8.2** (Möbius grid). *Let  $k, l \geq 1$ . Let  $G$  be the  $(k \times l)$ -grid. For any  $i \in \{1, \dots, k\}$ ,  $j \in \{1, \dots, l\}$ , let  $v_{i,j}$  be the vertex at the  $k$ -th row, and  $l$ -th column of  $G$ . Let  $H$  be the graph obtained from  $G$  by adding for every  $j \in \{1, \dots, l\}$ , the edge  $\{v_{1,l}, v_{k,l-j}\}$ . We call  $H$  the  $(k \times l)$ -Möbius grid.*

---

<sup>2</sup> We remark that there is a simple polynomial-time algorithm for testing orientability. Since we are dealing with cell-embeddings, this amounts to deciding whether there exists a way of orienting every face, so that the resulting orientations are consistent along edges. This can be done by first orienting an arbitrary face, and then continuing to adjacent faces. The algorithm terminates either when a globally-consistent orientation is found, certifying that the surface is orientable, or when we reach a face that cannot be consistently oriented (w.r.t. its adjacent faces), certifying that the surface contains a Möbius band, and is therefore nonorientable.

Lemma 8.1 implies the following.

**Corollary 8.3.** *Let  $k \geq 3$ , and let  $G$  be the  $(k \times k)$ -Möbius grid. Then,  $\text{genus}(G) = \Omega(k)$ .*

For two loops  $\gamma, \delta$  in some surface, and an integer  $t$ , we say that  $\gamma$  is  $t$ -freely homotopic to  $\delta$ , if  $\gamma$  is homotopic to  $\delta^t$ , where  $\delta^t$  denotes the loop obtained by concatenating  $t$  copies of  $\delta$ .

The following result is due to Brunet et. al [BMR96]. The precise formulation cited here, is implicit in their proof (proof of theorem 6.1 in [BMR96], with inductive invariant in step III.A).

**Lemma 8.4** (Brunet, Mohar, and Richter [BMR96]). *Let  $G$  be a graph, and let  $\varphi$  be a drawing of  $G$  into a nonorientable surface  $\mathcal{S}$ , with representativity  $\rho$ . Let  $\gamma$  be an orientation-reversing noose, of minimum length. Then,  $G$  contains a set of  $k = \lfloor (\rho - 1)/4 \rfloor$  disjoint pairwise homotopic cycles  $C_1, \dots, C_k$ , satisfying the following conditions:*

- (1) *For any  $i \in \{1, \dots, k\}$ , we have  $\varphi(C_i) \cap \gamma = \emptyset$ , and  $\varphi(C_i)$  is traversed by a loop 2-freely homotopic to  $\gamma$ .*
- (2) *For any  $i \in \{1, \dots, k\}$ , for every  $v \in V(C_i)$ , there exists  $w \in V(G)$ , with  $\varphi(w) \in \gamma$ , and a chain  $\xi$  with endpoints  $\varphi(v)$ , and  $\varphi(w)$ , such that  $\text{len}(\xi) \leq i + 1$ .*

The following lemma is the main technical ingredient required by our approximation algorithm of the orientable genus. It shows that any graph drawn into a nonorientable surface of nonorientable genus at least 2, and with large enough representativity, contains a large Möbius grid minor. This fact appears to be understood by certain experts. We give a formal proof for completeness. Perhaps something less obvious is that we actually obtain a polynomial dependence of the size of the Möbius grid minor, in terms of representativity. This polynomial dependence is necessary for our application.

**Lemma 8.5.** *Let  $G$  be a graph, and let  $\varphi$  be a drawing of  $G$  into a nonorientable surface  $\mathcal{S}$ , of nonorientable genus at least 2. Let  $\rho > 10$  be the representativity of  $\varphi$ . Then,  $G$  contains a  $(r \times r)$ -Möbius grid minor, for some  $r = \Omega(\sqrt{\rho})$ .*

*Proof.* Let  $\gamma$  be an orientation-reversing loop in  $\mathcal{S}$ , of minimum length (w.r.to  $\varphi$ ). Since  $\gamma$  is a noose, we have  $\rho' = \text{len}(\gamma) \geq \rho$ . Let  $C_1, \dots, C_k$  be the collection of cycles given by lemma 8.4, with  $k = \lfloor (\rho - 1)/4 \rfloor$ . There exists a neighborhood  $M$  of  $\gamma$ , homeomorphic to the Möbius strip, such that for any  $i \in \{1, \dots, k\}$ ,  $\varphi(C_i) \subset M$ . Cutting  $M$  along  $\gamma$ , we obtain a graph  $G'$  drawn into a cylinder  $M'$ . Let  $\varphi'$  be the induced drawing of  $G'$  on  $M'$ . The cylinder  $M'$ , has a boundary

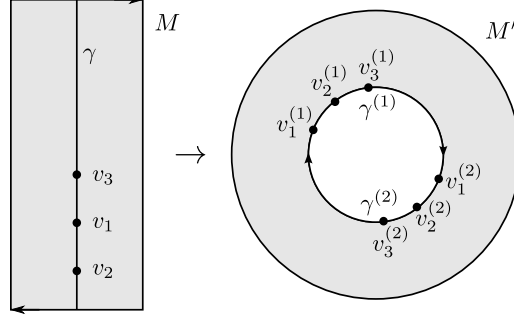
$$\gamma' = \gamma^{(1)} \cup \gamma^{(2)},$$

where for each  $\{1, 2\}$ ,  $\gamma^{(i)}$  is a segment corresponding to a copy of  $\gamma$ . Let

$$U = \{v \in V(G) : \varphi(v) \in \gamma\}.$$

For any  $v \in U$ , there exist two copies of  $v$  in  $V(G')$ ; for any  $i \in \{1, 2\}$ , let  $v^{(i)}$  be the copy of  $v$  with  $\varphi'(v^{(i)}) \in \gamma^{(i)}$ . Let also

$$U' = \{v^{(i)} : v \in U \text{ and } i \in \{1, 2\}\}.$$



Let  $t_1 = \sqrt{\rho}/10$ . For every  $w \in V(C_{t_1})$ , there exists  $v \in U$ , and a chain  $\xi_w$  in  $\varphi$ , between  $\varphi(w)$ , and  $\varphi(v)$ , with  $\text{len}(\xi_w) \leq t_1 + 1$ . By possibly shortcutting  $\xi_w$ , we can assume that it intersects  $\gamma$  only on  $\varphi(v)$ . Moreover, the collection of chains  $\{\xi_w\}_{w \in V(C_{t_1})}$  can be chosen so that no two curves intersect transversely. This can be done by observing that the length of chains induces a metric on  $V(G)$ , and by choosing the curves  $\xi_w$  to be shortest possible, breaking ties in a consistent fashion<sup>3</sup>.

Clearly, for every  $w \in V(C_{t_1})$ , we have  $\xi_w \subset M$ , since otherwise  $\xi_w$  has to intersect all the cycles  $C_{t_1+1}, \dots, C_k$ , and therefore its length cannot be at most  $t_1 + 1$ . Since  $\xi_w \subset M$ , and  $\xi_w$  intersects  $\gamma$  only on an endpoint  $\varphi(v)$ , for some  $v \in U$ , it follows that  $\xi_w$  lifts to a chain  $\xi'_w \subset M'$ , in  $\varphi'$ , with endpoints  $\varphi'(w)$ , and  $\varphi'(v^{(j)})$ , for some  $j \in \{1, 2\}$ .

Let  $W$  be a maximal subset  $W \subseteq V(C_t)$ , such that for any  $w \neq w' \in W$ , we have  $\xi'_w \cap \xi'_{w'} = \emptyset$ . Let  $t_2$  be the smallest integer such that for any set  $A \subset U'$  consisting of  $t_2$  consecutive vertices along the loop  $\gamma'$ , there exists some  $w \in W$ , with  $\xi'_w$  having an endpoint in  $A$ .

If  $t_2 \leq \sqrt{\rho}$ , then we can find the desired Möbius grid minor as follows. Let  $t_3 = \lfloor \rho'/(2t_2) \rfloor$ . Let  $v_1^{(1)}, \dots, v_{\rho'}^{(1)}, v_1^{(2)}, \dots, v_{\rho'}^{(2)}$  be an ordering of the vertices in  $U'$  induced by a traversal of  $\gamma'$ . For any  $i \in \{1, \dots, t_3\}$ , and  $j \in \{1, 2\}$ , let

$$A_i^{(j)} = \{v_{(i-1)t_3+1}^{(j)}, v_{(i-1)t_3+2}^{(j)}, \dots, v_{it_3}^{(j)}\}.$$

By planarity of  $M$  (see e.g. [CKS04]), we obtain a  $(t_1 \times 2t_3)$ -grid minor  $H'$  in  $G'$ , with minor mapping  $f : V(H') \rightarrow 2^{V(G')}$ , such that the vertices in the top row of  $H'$  are

$$f^{-1}(A_1^{(1)}), \dots, f^{-1}(A_{t_3}^{(1)}), f^{-1}(A_1^{(2)}), \dots, f^{-1}(A_{t_3}^{(2)}),$$

and in this order. Pulling  $f$  back to  $G$ , we arrive at a minor  $H$  in  $G$ , where  $H$  is obtained from  $H'$  by identifying for every  $i \in \{1, \dots, t_3\}$ , the pair of vertices  $f^{-1}(A_i^{(1)})$ , and  $f^{-1}(A_i^{(2)})$  of  $H'$ . It is straightforward to check that  $H$  contains a  $(t_4 \times t_4)$ -Möbius grid minor, where  $t_4 \geq \min\{t_1/2, t_3/2\} - 1 = \Omega(\min\{\sqrt{\rho}, \rho'/t_2\}) = \Omega(\min\{\sqrt{\rho}, \rho/t_2\})$ . Therefore, if  $t_2 \leq \sqrt{\rho}$ , then we are done.

It remains to consider the case  $t_2 > \sqrt{\rho}$ . We consider the following two subcases:

Case 1:  $|W| \geq 2$ . Pick  $w \neq w' \in W$ , such that the clockwise segment  $P$  of  $C_{t_1}$  between  $w$ , and  $w'$ , does not contain any other vertex  $w'' \in W \setminus \{w, w'\}$ . Let  $\varphi'(v)$ , and  $\varphi'(v')$ , be the endpoints of  $\xi'_w$ , and  $\xi'_{w'}$ , respectively in  $\gamma'$ , for some  $v, v' \in U'$ . By the assumption, we can pick  $w$ , and  $w'$ , so that the clockwise distance between  $v$ , and  $v'$  along  $\gamma'$  is at least  $t_2$ . Let  $P = w_1, \dots, w_\ell$ ,

<sup>3</sup>For example, we can redefine the length of a chain  $\xi$  to be the total *weighted cost* of the faces traversed by  $\xi$ , counting multiplicities, and where the weight of a face is set to  $1 + \varepsilon$ , for some perturbation  $\varepsilon = O(1/n^2)$ .

be that segment, where  $w_1 = w$ , and  $w_\ell = w'$ . By the maximality of  $W$ , it follows that for every  $i \in \{1, \dots, \ell\}$ , the chain  $\xi'_i$  intersects either  $\xi'_w$ , or  $\xi'_{w'}$  (or both). By planarity, it follows that there exists  $s \in \{1, \dots, \ell - 1\}$ , such that for any  $i \in \{1, \dots, s\}$ , we have

$$\xi'_{w_i} \cap \xi'_w \neq \emptyset,$$

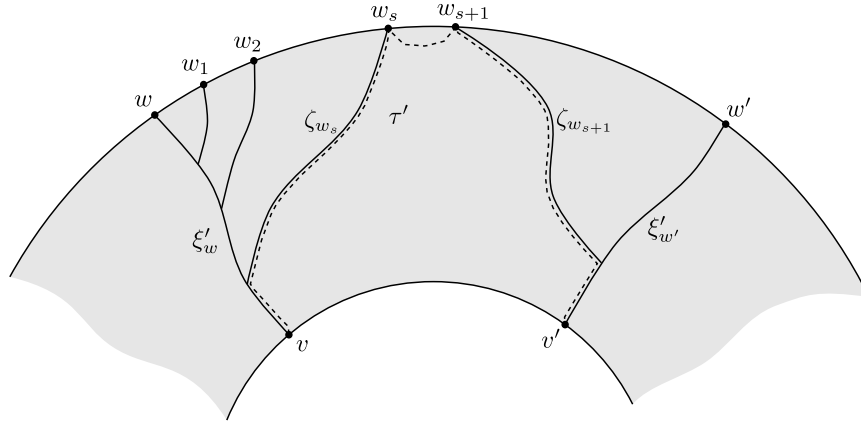
and for any  $j \in \{s + 1, \dots, \ell\}$ , we have

$$\xi'_{w_j} \cap \xi'_{w'} \neq \emptyset,$$

For any  $i \in \{1, \dots, s\}$ , let  $\zeta_{w_i}$  be the segment of  $\xi'_{w_i}$  between  $\varphi'(w_i)$ , and the first intersection point with  $\xi'_w$ . Similarly, for any  $j \in \{s + 1, \dots, \ell\}$ , let  $\zeta_{w_j}$  be the segment of  $\xi'_{w_j}$  between  $\varphi'(w_j)$ , and the first intersection point with  $\xi'_{w'}$ . Let  $\sigma$  be the subcurve of  $\xi'_w$  between  $\varphi'(v)$ , and the point  $\xi'_w \cap \zeta_{w_s}$ . Similarly, let  $\sigma'$  be the subcurve of  $\xi'_{w'}$  between  $\varphi'(v')$ , and the point  $\xi'_{w'} \cap \zeta_{w_{s+1}}$ . Since  $\{w_s, w_{s+1}\} \in E(G')$ , we can pick a chain  $\alpha$  between  $\varphi'(w_s)$ , and  $\varphi'(w_{s+1})$ , with  $\text{len}(\alpha) = 1$ . Let  $\tau'$  be the curve obtained by the concatenation

$$\tau' = \sigma \circ \zeta_{w_s} \circ \alpha \circ \zeta_{w_{s+1}} \circ \sigma',$$

where  $\circ$  denotes the usual concatenation of curves with a common endpoint.

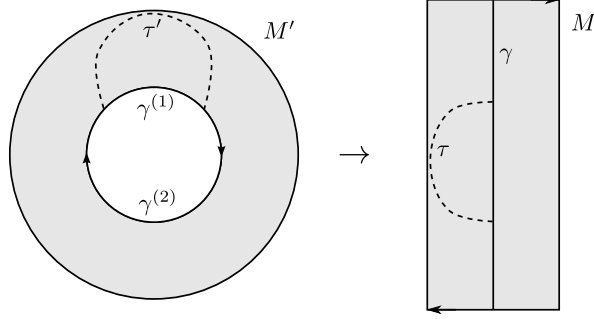


We have

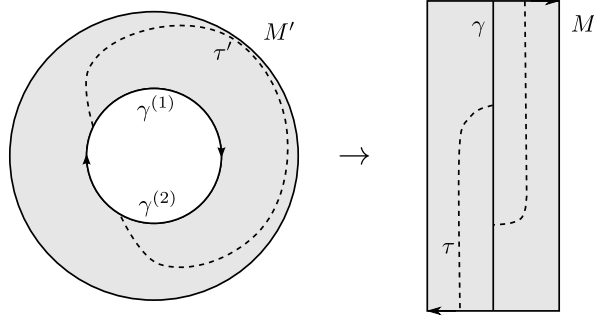
$$\begin{aligned} \text{len}(\tau') &= \text{len}(\sigma) + \text{len}(\zeta_{w_s}) + \text{len}(\alpha) + \text{len}(\zeta_{w_{s+1}}) + \text{len}(\sigma') \\ &\leq \text{len}(\xi'_w) + \text{len}(\xi'_{w_s}) + \text{len}(\alpha) + \text{len}(\xi'_{w_{s+1}}) + \text{len}(\xi'_{w'}) \\ &\leq 4t_1 + 5. \end{aligned}$$

Let  $v = v_i^{(j)}$ , and  $v' = v_{i'}^{(j')}$ , for some  $i, i' \in \{1, \dots, \rho'\}$ , and  $j, j' \in \{1, 2\}$ . Let  $\delta_1$ , and  $\delta_2$  be the two arcs of  $\gamma$  between  $v_i$ , and  $v_{i'}$ . Let  $\alpha$  be the clockwise distance between  $v$ , and  $v'$ , along  $\gamma'$ . We have the following two subcases:

Case 1.1: Assume  $\alpha < \rho'$ . In this case,  $\tau'$  corresponds to a chain  $\tau$  in  $\varphi$ . One of the two loops  $\tau \cup \delta_1$ , or  $\tau \cup \delta_2$ , is a noose homotopic to  $\gamma$ , of length  $\text{len}(\tau) + \rho' - t_2 < \rho'$ , which contradicts the fact that  $\gamma$  is the shortest orientation-reversing noose.



Case 1.2: Assume  $\alpha \geq \rho'$ . We have that  $\tau'$  corresponds to a chain  $\tau$  in  $\varphi$ , of length  $\text{len}(\tau) = \text{len}(\tau')$ . Consider the loops  $\eta_1 = \tau \cup \delta_1$ , and  $\eta_2 = \tau \cup \delta_2$ . Since  $\mathcal{S}$  has nonorientable genus at least 2, it follows that both  $\eta_1$ , and  $\eta_2$ , are noncontractible (see e.g. [BMR96]). One of these loops, say  $\eta_1$ , is an orientation-reversing noose, with  $\text{len}(\eta_1) = \text{len}(\tau) + \alpha - \rho'$ , and  $\eta_2$  is a non-contractible noose, which is 2-freely homotopic to  $\gamma$ , with  $\text{len}(\eta_2) = \text{len}(\tau) + 2\rho' - \alpha$ . If  $\alpha \leq 2\rho' + \text{len}(\tau) - \rho$ , then  $\text{len}(\eta_1) \leq \rho' + 2\text{len}(\tau) - \rho < \rho'$ , which contradicts the fact that  $\gamma$  is a shortest orientation-reversing noose in  $\varphi$ . Otherwise, if  $\alpha < 2\rho' + \text{len}(\tau) - \rho$ , then  $\text{len}(\eta_2) < \rho$ , which contradicts the fact that the representativity of  $\varphi$  is  $\rho$ .



Case 2:  $|W| = 1$ . Arguing as above, we construct a noncontractible noose in  $\varphi$ , of length at most  $4t_1 + 4 < \rho$ , which is a contradiction.

This concludes the proof.  $\square$

We are now ready to prove the main result of this section.

**Theorem 8.6** (Approximating the orientable genus). *There exists a polynomial-time algorithm which given a graph  $G$  of maximum degree  $\Delta$ , and an integer  $g > 0$ , either correctly decides that  $\text{genus}(G) > g$ , or outputs a drawing of  $G$  into a surface of orientable genus  $O(\Delta^4 g^9 \log^{13/2} n)$ .*

*Proof.* Using the algorithm from theorem 1.1, we can either correctly decide that  $\text{genus}(G) > g$ , or compute a drawing  $\varphi$  of  $G$  into a surface  $\mathcal{S}$  of Euler genus at most  $g' = O(\Delta^3 g^7 \log^{13/2} n)$ .

We inductively compute a sequence of graphs  $G_0, \dots, G_t$ , and for each  $i \in \{1, \dots, t\}$ , we also compute a drawing  $\varphi_i$  of  $G_i$  into some surface  $\mathcal{S}_i$ . Set  $G_0 = G$ ,  $\mathcal{S}_0 = \mathcal{S}$ , and  $\varphi_0 = \varphi$ . Given  $G_i$ ,  $\mathcal{S}_i$ , and  $\varphi_i$ , if  $\mathcal{S}_i$  is orientable, then we set  $t = i$ , and terminate the sequence at  $G_i$ . Otherwise, we compute  $G_{i+1}$ ,  $\mathcal{S}_{i+1}$ , and  $\varphi_{i+1}$  as follows. Let  $\rho_i$  be the representativity of  $\varphi_i$ . Let us assume first that  $\rho_i > \alpha g^2$ , for some constant  $\alpha > 0$ , to be determined later. If  $\mathcal{S}_i$  is the real projective plane,

then Lemma 8.1 implies that  $\text{genus}(G) > \lfloor \alpha g^2/2 \rfloor > g$ . In this case, we can therefore correctly decide that  $\text{genus}(G) > g$ , for  $\alpha \geq 4$ . Otherwise, we may assume that  $\mathcal{S}_i$  has nonorientable genus at least 2. In this case, we can apply Lemma 8.5 to obtain a  $(r \times r)$ -Möbius grid minor, for some  $r \geq \beta \sqrt{\alpha g^2}$ , for some universal constant  $\beta > 0$  (determined by the hidden constant in Lemma 8.5). By corollary 8.3, we have that  $\text{genus}(G) > \beta' \beta \sqrt{\alpha} g$ , for some universal constant  $\beta' > 0$ . By setting  $\alpha = 1/(\beta \beta')$ , we correctly decide that, in this case,  $\text{genus}(G) > g$ . Finally, it remains to consider the case  $\rho_i \leq \alpha g^2$ . We find a shortest noose  $\gamma_i$  in  $\varphi_i$  (this can be done in polynomial time, e.g. with the algorithm of Cabello et al. [CdVL10]). We set  $G_{i+1} = G_i \setminus X_i$ , where  $X_i$  is the set of vertices that  $\gamma_i$  intersects. We also set  $\mathcal{S}_{i+1}$  be the surface obtained by cutting  $\mathcal{S}_i$  along  $\gamma_i$ , and replacing any created punctures by disks. Note that the resulting surface might be disconnected, but this does not affect our algorithm. Let also  $\varphi_{i+1}$  be the induced drawing of  $G_{i+1}$  into  $\mathcal{S}_{i+1}$ .

Observe that since we arrive at  $\mathcal{S}_{i+1}$  by cutting  $\mathcal{S}_i$  along a noncontractible curve, it follows that the sequence  $G_1, \dots, G_t$  terminates at some finite  $t = O(g')$ . Let

$$X = \bigcup_{i=0}^{t-1} X_i,$$

i.e.  $X = V(G) \setminus V(G_t)$  is the set of all vertices removed by the algorithm. We have

$$\begin{aligned} |X| &= \sum_{i=0}^{t-1} \rho_i \\ &\leq t \alpha g^2 \\ &= O(\Delta^3 g^9 \log^{13/2} n). \end{aligned}$$

Since  $\mathcal{S}_t$  is orientable, we can obtain a drawing of  $G$  into an orientable surface  $\mathcal{S}'$  by adding for every edge incident to a vertex in  $X$ , a new handle in  $\mathcal{S}_t$ . This results into a drawing of  $G$  into an orientable surface  $\mathcal{S}'$  of orientable genus

$$\begin{aligned} \text{genus}(\mathcal{S}') &= \text{genus}(\mathcal{S}_t) + \Delta \cdot |X| \\ &= O(\text{genus}(\mathcal{S}) + \Delta \cdot |X|) \\ &= O(\Delta^4 g^9 \log^{13/2} n), \end{aligned}$$

as required. □

## 9 Approximating the crossing number

In this section we give an algorithm for approximating the crossing number of a graph by building on our algorithm for approximating the orientable genus of a graph. We remark that we can obtain a similar algorithm for crossing number by instead using our algorithm for Euler genus. However, using orientable drawings simplifies the analysis.

**Definition 9.1** (Augmented cylinder). *Let  $r, t \geq 1$ , and let  $G$  be the  $(r \times t)$ -cylinder. Let  $G'$  be the graph obtained from  $G$  by adding the edge  $\{u, v\}$ , between some vertex  $u$  in the top of  $G$ , and some vertex  $v$  in the bottom of  $G$ . We say that  $G'$  is an  $(r \times t)$ -augmented cylinder.*

The following fact is not difficult, it can be proven e.g. using the arguments from [CMS11].



**Lemma 9.2.** *Let  $r \geq 3$ , and let  $G$  be an  $(r \times r)$ -augmented cylinder. Then, the crossing number of  $G$  is  $\Omega(r)$ .*

We remark that the same bound holds on the crossing number of an  $(r \times 3)$ -augmented cylinder. This refined statement however, does not make any difference in our application.

We now show that any graph that admits a “dense enough” drawing into an orientable surface of positive orientable genus, contains a large augmented cylinder minor. It seems that one should also be able to get a large toroidal minor, and in fact such a result has been obtained for graphs drawn into the torus by Schrijver [Sch93], and by Hliněný, and Salazar [HS07]. It is possible that an extension of these results to higher genus surfaces can be obtained by the techniques in [BMR96]. This however does not seem immediate. Let us also point out that such an improvement would immediately yield an improved constant in the exponent of the approximation guarantee of our algorithm.

We will use the following results to construct the desired augmented cylinder.

**Lemma 9.3** (Brunet et al. [BMR96]). *Let  $G$  be a graph, and let  $\varphi$  be a drawing of  $G$  into an orientable surface  $\mathcal{S}$ , of positive orientable genus. Let  $\rho$  be the representativity of  $\varphi$ . Then,  $G$  contains a set of  $\lfloor(\rho - 1)/2\rfloor$  pairwise disjoint, pairwise freely homotopic nonseparating cycles.*

**Lemma 9.4** (Brunet et al. [BMR96]). *Let  $G$  be a graph, and let  $\varphi$  be a drawing of  $G$  into a surface, with representativity  $\rho$ . Let  $C, C'$  be disjoint homotopic noncontractible cycles in  $G$ . Then,  $G$  contains  $\rho$  pairwise disjoint paths, each contained within the cylinder bounded by  $C$ , and  $C'$ , and each having one end in  $C$ , and the other in  $C'$ .*

We now prove that large representativity in a orientable surface implies the existence of a large augmented cylinder minor.

**Lemma 9.5.** *Let  $G$  be a graph, and let  $\varphi$  be a drawing of  $G$  into an orientable surface  $\mathcal{S}$ , with orientable genus  $g > 0$ . Let  $\rho \geq 3$  be the representativity of  $\varphi$ . Then,  $G$  contains a  $(r \times r)$ -augmented cylinder minor, for some  $r = \Omega(\rho)$ .*

*Proof.* By Lemma 9.3, there exists a collection  $C_1, \dots, C_k$  of pairwise disjoint, pairwise freely homotopic nonseparating cycles in  $G$ , with  $k = \Omega(\rho)$ . By a possible reordering of the indices, we have that there exists a cylinder  $\mathcal{C} \subset \mathcal{S}$ , with boundaries  $C_1$ , and  $C_k$ . By Lemma 9.4, there exists a collection  $P_1, \dots, P_\ell$  of pairwise disjoint paths in  $\mathcal{C}$ , each having an endpoint in  $C_1$ , and an endpoint in  $C_k$ , with  $\ell = \lfloor(\rho - 1)/2\rfloor$ . The graph  $H = \left(\bigcup_{i=1}^k C_i\right) \cup \left(\bigcup_{j=1}^\ell P_j\right)$  contains a  $(k \times \ell)$ -cylinder minor. Since the cycles  $C_1, \dots, C_k$  are nonseparating, it follows that  $\mathcal{S} \setminus \mathcal{C}$  is a connected surface. This implies that there exists a path  $Q$  in  $G$  between  $C_1$ , and  $C_k$ , that intersects  $\mathcal{C}$  only on its endpoints. Contracting  $Q$  into an edge, gives a  $(k \times \ell)$ -augmented cylinder minor, as required.  $\square$

We also need the following result due to Garcia-Moreno, and Salazar [GMS01], which allows us to bound the crossing number of a graph  $G$ , in terms of the crossing number of a minor of  $G$ , with maximum degree 4.

**Lemma 9.6** (Garcia-Moreno, and Salazar [GMS01]). *Let  $G$  be a graph, and let  $H$  be a minor in  $G$ , of maximum degree 4. Then,  $\text{cr}(G) \geq \text{cr}(H)/4$ .*

The following result allows us to reduce the problem of approximating the crossing number, to the problem of computing a set of edges of minimum cardinality, whose removal leaves a planar graph. The first result of this type was obtained in [CMS11], but the following gives a slightly better dependence on the maximum degree.

**Lemma 9.7** (Chimani, and Hlinený [CH11]). *There exists a polynomial-time algorithm which given a graph  $G$  of maximum degree  $\Delta$ , and a set  $E^* \subseteq E(G)$ , with  $|E^*| = \ell$ , such that  $G \setminus E^*$  is planar, outputs a drawing of  $G$  into the plane, with at most  $O(\Delta \cdot \ell \cdot (\ell + \text{cr}(G)))$  crossings.*

We are now ready to present our approximation algorithm for crossing number.

**Theorem 9.8.** *There exists a polynomial-time algorithm which given a graph  $G$  of maximum degree  $\Delta$ , and an integer  $k \geq 0$ , either correctly decides that  $\text{cr}(G) > k$ , or outputs a drawing of  $G$  into the plane with at most  $O(\Delta^{11} k^{20} \log^{13} n)$  crossings.*

*Proof.* If  $G$  is planar, then there is nothing to be done. We may therefore assume that  $G$  is not planar.

Recall that  $\text{genus}(G) \leq \text{cr}(G)$ . Therefore, using the algorithm from Theorem 8.6, we can either correctly decide that  $\text{cr}(G) > k$ , or we can compute a drawing  $\varphi$  of  $G$  into an orientable surface  $\mathcal{S}$  of orientable genus  $g = O(\Delta^4 k^9 \log^{13/2} n)$ . We now proceed to inductively compute a sequence of subgraphs  $G = G_0 \supset \dots \supset G_t$ . For every  $i \in \{0, \dots, t\}$ , we also compute a drawing  $\varphi_i$  of  $G_i$  into some orientable surface  $\mathcal{S}_i$ . Set  $\varphi_0 = \varphi$ , and  $\mathcal{S}_0 = \mathcal{S}$ .

Given  $G_i$ ,  $\varphi_i$ , and  $\mathcal{S}_i$ , we proceed as follows. If  $G_i$  is planar, then we set  $t = i$ , and we terminate the sequence at  $G_i$ . Otherwise, let  $\rho_i$  be the representativity of  $\varphi_i$ . We find a shortest nonseparating noose  $\gamma_i$  in  $\varphi_i$  (using the algorithm from [CdVL10]). If  $\text{len}(\gamma_i) = \rho_i > \alpha k$ , for some universal constant  $\alpha > 0$  (to be determined later), then by Lemma 9.5 we find a  $(r \times r)$ -augmented cylinder minor in  $G_i \subseteq G$ , with  $r > \beta \alpha k$ , for some universal constant  $\beta$ . Combining lemmas 9.2, and 9.6, we obtain that  $\text{cr}(G) \geq \text{cr}(G_i) > \beta' \beta \alpha k$ , for some universal constant  $\beta' > 0$ . By setting  $\alpha = 1/(\beta \beta')$ , this leads to a proof that  $\text{cr}(G) > k$ . Therefore, it remains to consider the case  $\text{len}(\gamma_i) \leq \alpha k$ . In this case, we set  $G_{i+1} = G_i \setminus X_i$ , where  $X_i$  is the set of vertices in  $\gamma_i$ . We also set  $\mathcal{S}_{i+1}$  to be the surface obtained from  $\mathcal{S}_i$  by cutting along  $\gamma_i$ , and placing disks on any possible created punctures. Let also  $\varphi_{i+1}$  be the induced drawing of  $G_{i+1}$  into  $\mathcal{S}_{i+1}$ . This completes the definition of the sequence  $G_0, \dots, G_t$ . We argue that the sequence terminates at some finite  $t$ . Indeed, we can only cut along at most  $O(g)$  nooses before a surface becomes trivial (cutting along a separating noose disconnects the surface, but this does not affect our argument). Therefore,  $t = O(g)$ .

Let  $X = \bigcup_{i=0}^{t-1} X_i$ , and let  $E^*$  be the set of all edges incident to vertices in  $X$ . We have

$$\begin{aligned} |E^*| &\leq \Delta \cdot |X| \\ &= \Delta \cdot \sum_{i=1}^t \rho_i \\ &\leq \Delta t \alpha k \\ &= O(\Delta^5 k^{10} \log^{13/2} n). \end{aligned}$$

Since  $G \setminus E^*$  is a planar graph, using the algorithm from Lemma 9.7, we obtain in polynomial time a drawing of  $G$  into the plane, with at most  $O(\Delta \cdot |E^*| \cdot (|E^*| + k)) = O(\Delta^{11} k^{20} \log^{13} n)$  crossings, as required.  $\square$

## 10 Approximating planar edge and vertex deletion

In this Section we give approximation algorithms for the minimum planar vertex/edge deletion problems. Recall that for a graph  $G$ ,  $\text{vertex-planarization}(G)$  denotes the minimum cardinality of a set  $X \subset V(G)$ , such that  $G \setminus X$  is planar. Similarly,  $\text{edge-planarization}(G)$  denotes the minimum cardinality of a set  $Y \subset E(G)$ , such that  $G \setminus Y$  is planar. We first derive an algorithm for  $\text{vertex-planarization}(G)$  and then via the degree bound assumption an algorithm for  $\text{edge-planarization}(G)$ .

We start with a lemma by Mohar that shows that a sufficiently large representativity of an embedding certifies a bound on the genus.

**Lemma 10.1** (Mohar [Moh92]). *Let  $G$  be a graph, and let  $\varphi$  be a drawing of  $G$  into a surface of positive orientable genus, with representativity  $\rho > 2\text{genus}(G) + 2$ . Then,  $\text{genus}(S) = \text{genus}(G)$ .*

The following easy lemma shows that the representativity of an embedding can be lower bounded after the removal of a set of vertices.

**Lemma 10.2.** *Let  $G$  be a graph, and let  $\varphi$  be a drawing of  $G$  into an orientable surface  $S$  of orientable genus  $g > 0$ , with representativity  $\rho$ . Let  $X \subset V(G)$ , and let  $G' = G \setminus X$ . Let  $\varphi'$  be the drawing of  $G'$  into  $S$  obtained by restricting  $\varphi$  to  $G'$ , and let  $\rho'$  be the representativity of  $\varphi'$ . Then,  $\rho' \geq \rho - |X|$ .*

*Proof.* Let  $\gamma'$  be a noose in  $\varphi'$ . We can modify  $\gamma'$  to obtain a noose  $\gamma$  in  $\varphi$  as follows. Consider a face  $F \subset G'$  in  $\varphi'$  that is visited by  $\gamma$ , and suppose that  $F$  is not a face in  $\varphi$ . Let  $Z \subseteq X$  be the set of vertices in  $G$  with an image in the disk  $\mathcal{D}$  bounded by  $F$ . We shortcut the noose  $\gamma'$ , so that inside  $\mathcal{D}$  it intersects  $\varphi(G)$  only on  $\varphi(Z)$ . This can clearly be done by increasing the length of the noose by at most  $|Z|$ . We repeat the same process for all faces visited by  $\gamma'$ . The resulting curve  $\gamma$  is a noose in  $\varphi$ , and with total length at most  $\text{len}(\gamma') + |X|$ . This implies that  $\rho' \geq \rho - |X|$ , as required.  $\square$

Now we obtain our algorithm for  $\text{vertex-planarization}(G)$  encapsulated in the theorem below.

**Theorem 10.3** (Approximating the minimum planar vertex deletion). *There exists a polynomial-time algorithm which given a graph  $G$  of maximum degree  $\Delta$ , and an integer  $k > 0$ , either correctly decides that  $\text{vertex-planarization}(G) > k$ , or outputs a set  $X \subseteq V(G)$ , with  $|X| = O(\Delta^5 k^{10} \log^{13/2} n)$ , such that  $G \setminus X$  is planar.*

*Proof.* The algorithm is similar to the one used in Theorem 9.8 for approximating the crossing number of a graph. We may assume that  $G$  is not planar, since otherwise the assertion is trivially true.

We have that  $\text{genus}(G) \leq \Delta \cdot \text{vertex-planarization}(G)$ . Therefore, using the algorithm from Theorem 8.6, we can either correctly decide that  $\text{vertex-planarization}(G) > k$ , or we can compute a drawing  $\varphi$  of  $G$  into an orientable surface  $S$  of orientable genus  $g = O(\Delta^4 k^9 \log^{13/2} n)$ . We next inductively compute a sequence of subgraphs  $G = G_0 \supset \dots \supset G_t$ . For every  $i \in \{0, \dots, t\}$ , we also compute a drawing  $\varphi_i$  of  $G_i$  into some orientable surface  $S_i$ . Set  $\varphi_0 = \varphi$ , and  $S_0 = S$ .

Given  $G_i$ ,  $\varphi_i$ , and  $S_i$ , we proceed as follows. If  $G_i$  is planar, then we set  $t = i$ , and we terminate the sequence at  $G_i$ . Otherwise, let  $\rho_i$  be the representativity of  $\varphi_i$ . We find a shortest nonseparating noose  $\gamma_i$  in  $\varphi_i$  (using the algorithm from [CdVL10]).

Suppose first that  $\text{len}(\gamma_i) = \rho_i > (2\Delta + 1)k + 2$ . Assume that  $\text{vertex-planarization}(G) \leq k$ . Then, there exists  $X_{\text{OPT}} \subset V(G)$ , with  $|X_{\text{OPT}}| \leq k$ , and such that the graph  $G' = G \setminus X_{\text{OPT}}$  is planar. Let  $\varphi'$  be the drawing of  $G'$  into  $\mathcal{S}$  induced by restricting  $\varphi$  to  $G'$ , and let  $\rho'$  be the representativity of  $\varphi'$ . By Lemma 10.2 we have

$$\begin{aligned} \rho' &\geq \rho_i - k \\ &> 2 \cdot \Delta \cdot k + 2 \\ &= 2 \cdot \Delta \cdot \text{vertex-planarization}(G) + 2 \\ &\geq 2 \cdot \text{genus}(G) + 2 \\ &\geq 2 \cdot \text{genus}(G') + 2. \end{aligned}$$

Combining with Lemma 10.1 we deduce that  $\text{genus}(G') = g > 0$ , i.e. that  $G'$  is not planar, which is a contradiction. Therefore, if  $\rho_i > (2\Delta + 1)k + 2$ , we obtain a proof that  $\text{vertex-planarization}(G) > k$ .

It remains to consider the case  $\text{len}(\gamma_i) = \rho_i \leq (2\Delta + 1)k + 2$ . In this case, we set  $G_{i+1} = G_i \setminus X_i$ , where  $X_i$  is the set of vertices in  $\gamma_i$ . We also set  $\mathcal{S}_{i+1}$  to be the surface obtained from  $\mathcal{S}_i$  by cutting along  $\gamma_i$ , and placing disks on any possible created punctures. Let also  $\varphi_{i+1}$  be the induced drawing of  $G_{i+1}$  into  $\mathcal{S}_{i+1}$ . This completes the definition of the sequence  $G_0, \dots, G_t$ . As in the proof of Theorem 9.8, we have  $t = O(g)$ .

We set  $X = \bigcup_{i=0}^{t-1} X_i$ . We have

$$\begin{aligned} |X| &\leq \sum_{i=1}^t \rho_i \\ &\leq t(2\Delta + 1)k + 2 \\ &= O(\Delta^5 k^{10} \log^{13/2} n). \end{aligned}$$

By construction, the graph  $G_t = G \setminus X$  is planar, which concludes the proof.  $\square$

Observe that for any graph  $G$  of maximum degree  $\Delta$ , the parameters  $\text{edge-planarization}(G)$ , and  $\text{vertex-planarization}(G)$  can differ by at most a multiplicative factor of  $\Delta$ . Combining this simple observation with Theorem 10.3, we immediately obtain the following result.

**Theorem 10.4** (Approximating the minimum planar edge deletion). *There exists a polynomial-time algorithm which given a graph  $G$  of maximum degree  $\Delta$ , and an integer  $k > 0$ , either correctly decides that  $\text{edge-planarization}(G) > k$ , or outputs a set  $Y \subseteq E(G)$ , with  $|Y| = O(\Delta^6 k^{10} \log^{13/2} n)$ , such that  $G \setminus Y$  is planar.*

**Acknowledgments:** We thank Jeff Erickson and Ken-ichi Kawarabayashi for useful discussions.

## References

- [BHK62] Joseph Battle, Frank Harary, and Yukihiro Kodama. Additivity of the genus of a graph. *Bull. Amer. Math. Soc.*, 68(6):565–568, 1962.
- [BL84] Sandeep N. Bhatt and Frank Thomson Leighton. A framework for solving vlsi graph layout problems. *J. Comput. Syst. Sci.*, 28(2):300–343, 1984.

- [BMR96] Richard Brunet, Bojan Mohar, and R. Bruce Richter. Separating and nonseparating disjoint homotopic cycles in graph embeddings. *J. Comb. Theory, Ser. B*, 66(2):201–231, 1996.
- [CdVL10] Sergio Cabello, Éric Colin de Verdière, and Francis Lazarus. Finding shortest non-trivial cycles in directed graphs on surfaces. In *Symposium on Computational Geometry*, pages 156–165, 2010.
- [CH11] Markus Chimani and Petr Hlinený. A tighter insertion-based approximation of the crossing number. In *ICALP (1)*, pages 122–134, 2011.
- [Chu11] Julia Chuzhoy. An algorithm for the graph crossing number problem. In *STOC*, pages 303–312, 2011.
- [CKK97] Jianer Chen, Saroja P. Kanchi, and Arkady Kanevsky. A note on approximating graph genus. *Inf. Process. Lett.*, 61(6):317–322, 1997.
- [CKS04] Chandra Chekuri, Sanjeev Khanna, and F. Bruce Shepherd. Edge-disjoint paths in planar graphs. In *FOCS*, pages 71–80, 2004.
- [CMS11] Julia Chuzhoy, Yury Makarychev, and Anastasios Sidiropoulos. On graph crossing number and edge planarization. In *SODA*, pages 1050–1069, 2011.
- [DFHT05] Erik D. Demaine, Fedor V. Fomin, Mohammad Taghi Hajiaghayi, and Dimitrios M. Thilikos. Subexponential parameterized algorithms on bounded-genus graphs and  $h$ -minor-free graphs. *J. ACM*, 52(6):866–893, 2005.
- [Epp13] David Eppstein. Grid Minors in Damaged Grids. *CoRR*, abs/1303.1136, March 2013.
- [ES13] Jeff Erickson and Anastasios Sidiropoulos. A near-optimal approximation algorithm for asymmetric TSP on embedded graphs. Manuscript, 2013.
- [FHL08] Uriel Feige, MohammadTaghi Hajiaghayi, and James R. Lee. Improved approximation algorithms for minimum weight vertex separators. *SIAM J. Comput.*, 38(2):629–657, 2008.
- [FHRR95] J. R. Fiedler, J. P. Huneke, R. B. Richter, and N. Robertson. Computing the orientable genus of projective graphs. *Journal of Graph Theory*, 20(3):297–308, 1995.
- [GMS01] Enrique García-Moreno and Gelasio Salazar. Bounding the crossing number of a graph in terms of the crossing number of a minor with small maximum degree. *Journal of Graph Theory*, 36(3):168–173, 2001.
- [GT01] J.L. Gross and T.W. Tucker. *Topological graph theory*. Dover Publications, 2001.
- [GT11] Qian-Ping Gu and Hisao Tamaki. Constant-factor approximations of branch-decomposition and largest grid minor of planar graphs in  $O(n^{1+\epsilon})$  time. *Theor. Comput. Sci.*, 412(32):4100–4109, 2011.
- [HS07] Petr Hlinený and Gelasio Salazar. Approximating the crossing number of toroidal graphs. In *ISAAC*, pages 148–159, 2007.

- [HT74] John Hopcroft and Robert Tarjan. Efficient planarity testing. *J. ACM*, 21(4):549–568, October 1974.
- [KMR08] Kenichi Kawarabayashi, Bojan Mohar, and Bruce A. Reed. A simpler linear time algorithm for embedding graphs into an arbitrary surface and the genus of graphs of bounded tree-width. In *FOCS*, pages 771–780, 2008.
- [KR07] Kenichi Kawarabayashi and Bruce A. Reed. Computing crossing number in linear time. In *STOC*, pages 382–390, 2007.
- [LR99] Frank Thomson Leighton and Satish Rao. Multicommodity max-flow min-cut theorems and their use in designing approximation algorithms. *J. ACM*, 46(6):787–832, 1999.
- [MNS12] Yury Makarychev, Amir Nayyeri, and Anastasios Sidiropoulos. A pseudo-approximation algorithm for the genus of hamiltonian graphs. Manuscript, 2012.
- [Moh92] Bojan Mohar. Combinatorial local planarity and the width of graph embeddings. *Canad. J. Math.*, 44(6):1272–1288, 1992.
- [Moh99] Bojan Mohar. A linear time algorithm for embedding graphs in an arbitrary surface. *SIAM J. Discrete Math.*, 12(1):6–26, 1999.
- [Moh01] B. Mohar. Face covers and the genus problem for apex graphs. *Journal of Combinatorial Theory, Series B*, 82(1):102–117, 2001.
- [MT01] B. Mohar and C. Thomassen. *Graphs on surfaces*. Johns Hopkins University Press, 2001.
- [OS11] Shayan Oveis Gharan and Amin Saberi. The asymmetric traveling salesman problem on graphs with bounded genus. In *SODA*, pages 967–975, 2011.
- [RS86] Neil Robertson and Paul D. Seymour. Graph minors. V. Excluding a planar graph. *J. Comb. Theory, Ser. B*, 41(1):92–114, 1986.
- [RS90] Neil Robertson and Paul D. Seymour. Graph minors. VIII. A Kuratowski theorem for general surfaces. *J. Comb. Theory, Ser. B*, 48(2):255–288, 1990.
- [RS03] Neil Robertson and Paul D. Seymour. Graph minors. XVI. Excluding a non-planar graph. *J. Comb. Theory, Ser. B*, 89(1):43–76, 2003.
- [RST94] Neil Robertson, Paul D. Seymour, and Robin Thomas. Quickly excluding a planar graph. *J. Comb. Theory, Ser. B*, 62(2):323–348, 1994.
- [Sch93] Alexander Schrijver. Graphs on the torus and geometry of numbers. *J. Comb. Theory, Ser. B*, 58(1):147–158, 1993.
- [ST94] Paul D. Seymour and Robin Thomas. Call routing and the ratcatcher. *Combinatorica*, 14(2):217–241, 1994.
- [Tho89] Carsten Thomassen. The graph genus problem is np-complete. *J. Algorithms*, 10(4):568–576, 1989.

- [Tho93] Carsten Thomassen. Triangulating a surface with a prescribed graph. *J. Comb. Theory, Ser. B*, 57(2):196–206, 1993.
- [Whi01] A.T. White. *Graphs of groups on surfaces: interactions and models*. Elsevier, 2001.